RICE UNIVERSITY

Space–Time FSI Modeling and Dynamical Analysis of Spacecraft Parachutes and Parachute Clusters

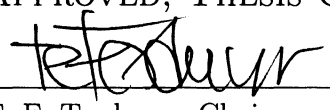
by

Timothy R. Spielman, 2nd Lt, USAF

A Thesis Submitted
in Partial Fulfillment of the
Requirements for the Degree

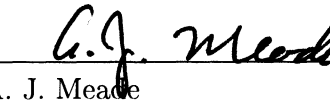
Master of Science

Approved, Thesis Committee:

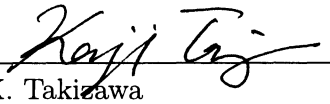


T. E. Tezduyar, Chair
Professor of Mechanical Engineering and
Materials Science

J. E. Akin
Professor of Mechanical Engineering and
Materials Science and Professor of
Computational and Applied Mathematics



A. J. Meade
Professor of Mechanical Engineering and
Materials Science



K. Takizawa
Associate Professor in Department of
Modern Mechanical Engineering and
Waseda Institute for Advanced Study
Waseda University, Tokyo, Japan

Houston, Texas
April 2011

The views expressed in this thesis are those of the author and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U. S. Government.

Abstract

Space–Time FSI Modeling and Dynamical Analysis of Spacecraft Parachutes and Parachute Clusters

by

Timothy R. Spielman

The Team for Advanced Flow Simulation and Modeling (T★AFSM) at Rice University has been developing fluid–structure interaction (FSI) modeling techniques using Stabilized Space–Time FSI (SSTFSI) core technology to model spacecraft parachutes and carry out informative dynamical analysis of parachute performance. Computer modeling of spacecraft parachutes, which are quite often used in clusters of two or three large parachutes, involves FSI between the parachute canopy and the air, geometric complexities created by the construction of the ringsail parachute with hundreds of gaps and slits, and the contact between the parachutes. The computational challenges related to the FSI have successfully been addressed, and one of the special techniques used to deal with the geometric complexities is the Homogenized Modeling of Geometric Porosity. The technique for modeling, in the context of an FSI problem, the contact between two structural surfaces is described and the results of FSI computations using this technique are presented. The results obtained from FSI computations of single parachutes and parachute clusters, the related dynamical analysis, and a special decomposition technique for parachute descent speed are presented. A special technique for extracting model parameters from a parachute FSI computation is also presented.

Acknowledgments

I would like to thank Dr. Tayfun Tezduyar for offering me the opportunity to come to Rice and pursue research as part of the T★AFSM. His guidance and advice during my time at Rice has been invaluable. I would also like to thank Dr. Kenji Takizawa, who is one of the most intelligent and hardest-working people I know. His tireless efforts both in and outside of the lab made this research possible. I have enjoyed interacting with him both professionally and personally and I wish him well in his new role as Associate Professor.

I thank my family for supporting and encouraging me in all of my endeavors. Without them, I would not be where I am today. Mom and Dad, thank for you shaping me into the man I have become and instilling in me the importance of hard work. To my siblings, Andrew and Annie, thank you for your never-ending love and support. And to my extended family and close friends, thank you for always being there for me. I am so grateful for all of you.

I would also like to thank all of the current and former members of the T★AFSM whom I have had the privilege of working with during my time at Rice: Creighton Moorman, Eddie Wright, Tyler Brummer, Brad Henicke, Tracee Curlett, Tony Puntel, Darren Montes, Kat Schjodt, Matt Guertin, Matt Fritze, and Nik Kostov. Your collaboration and insight have greatly contributed to this research. I have enjoyed your friendship during my time in Houston.

I want to thank everyone at NASA Johnson Space Center who met with me and

provided assistance. I would especially like to thank Mr. Gary Bourland, Mr. Ricardo Machin, and Mr. Jay LeBeau. Our discussions provided perspective and motivation for me to succeed in this research. Finally, I would like to thank Dr. Akin and Dr. Meade for the time they have invested as members of my thesis committee. This work was supported in part by NASA Johnson Space Center under grant NNX09AM89G. It was also supported in part by the Rice Computational Research Cluster funded by NSF under Grant CNS-0821727.

Contents

List of Figures

Chapter 1

Introduction

Many physical systems involve interactions between fluids and structures. Flows can be internal or external, and structures can range from nearly-rigid objects like buildings to very light and highly deformable materials like fabric parachutes. Given the shape of a structure in a flowfield, the velocity and pressure of the fluid moving around it can be calculated at points in space and time. From the flowfield, fluid traction forces acting on the structure can be calculated. These fluid traction forces cause the structure to deform, and the deformation consequently alters the flowfield. Because the flowfield is dependent on the shape of the structure and the structural deformation is dependent on the fluid forces, the fluid and structural unknowns are coupled and the system of governing equations must be solved simultaneously. The two-way interaction between fluid and structure occurs continuously and is even more pronounced in the case of light materials where the structure is very sensitive to changes in fluid forces. This class of problem is commonly referred to as fluid–structure interaction (FSI). Obtaining solutions for FSI problems is inherently challenging because the fluid and structural mechanics are governed by fundamentally different nonlinear, time-dependent partial differential equations that must be solved simultaneously with matching interface conditions.

FSI problems are generally too complex to solve analytically. For example, exact solutions to the Navier–Stokes equations are usually limited to specific cases with simple geometry. Experimental approaches can be helpful for analysis, but they have several disadvantages as well. Testing can be expensive, and some data may be difficult to measure during experiments. The testing environment is often difficult to control, and in the case of large parachutes, the desired test conditions may be difficult or impossible to achieve in a laboratory. Numerical approaches can be faster and less costly than experimentation, and since the problem is solved at discrete points, parameters of interest can be calculated virtually anywhere in the computational domain during post-processing. For these reasons, reliable numerical methods can be used for in-depth analysis of real-world FSI problems to complement laboratory and field testing.

The Team for Advanced Flow Simulation and Modeling (T★AFSM) at Rice University has been developing FSI modeling techniques for ringsail parachutes for several years. These are typically used in clusters of two or three parachutes. The two major computational challenges successfully addressed so far have been the FSI between the parachute canopy and the air and the geometric complexities created by the construction of the ringsail parachute with hundreds of gaps and slits. The contact between the parachutes creates another major challenge, which the T★AFSM started addressing very recently (see [34]).

The research in this thesis expands upon the foundation of advanced parachute modeling techniques developed by the T★AFSM to further enhance parachute cluster modeling capabilities. New dynamical analysis techniques for parachute FSI computations are also presented. These include a method for extracting model parameters from FSI computations and a special decomposition technique for informative parachute performance analysis.

1.1 Motivation

The parachute chosen for modeling in this research is expected to be used for the recovery of NASA's Orion Crew Exploration Vehicle (CEV) Crew Module. In 2004, the United States announced the Vision for Space Exploration, a new space policy for extending human presence across the solar system including a return to the Moon and later missions to Mars [19]. To pursue these goals, NASA established the Constellation Program and initiated the design of the Orion CEV, the Ares I crew launch vehicle, and the Ares V cargo launch vehicle [22]. The Constellation Program was canceled in late 2010, but development of the Orion CEV was directed to continue [6]. Orion is expected to become NASA's next generation crew vehicle that will restore the capability for low Earth orbit missions after the Space Shuttle is retired [21, 20]. The first unmanned test flight for Orion could happen as early as 2013, and the first manned flight is anticipated in 2016 [27, 6]. Subsequent flights could provide logistical support for the International Space Station if commercial orbital transport services are not available. The Orion CEV may eventually become a key component of human missions beyond low Earth orbit to the Moon, asteroids and Mars [20].

1.2 Parachute System Description

Spacecraft parachutes have four main components: canopy, suspension lines, riser, and payload. Canopy design can depend on many factors including payload weight, desired stability characteristics, and descent speed requirements. The canopy must be strong enough to withstand the aerodynamic forces required to produce sufficient drag, but the overall weight and storage volume of the parachute system must be small enough to facilitate practical integration with the spacecraft design. To accomplish this, the canopy is made of thin, lightweight material, and the load-bearing strength is provided by heavier suspension lines and riser elements. Most spacecraft parachutes

have an opening at the canopy apex, called a vent, to increase stability. Radial lines start at the vent and continue to the bottom of the canopy, which is also called the skirt. The slice of canopy between two adjacent radial lines is referred to as a gore. At the skirt, each radial line is connected to a suspension line. The suspension lines meet at a confluence point and connect to the riser. For a single main parachute, the payload is directly connected to the suspension line confluence by the riser. For a cluster configuration, the risers of the individual parachutes meet at a confluence, and a harness connects the payload with the riser confluence. The Orion CEV single main parachute assembly is shown in Figure 1.1.

Figure 1.1: Orion CEV single main parachute assembly.

Ringsail parachutes were used for spacecraft recovery in the Apollo program and have historically shown good reliability, drag efficiency, and damage tolerance [16]. Ringsail parachutes consist of rings and sails that create hundreds of gaps and slits in the canopy. The rings are located near the vent and are separated by gaps. The sails start below the rings and continue to the skirt. Sails are manufactured so that the

leading edge is longer than the distance between the radials. This creates "fullness" on the leading edge and slits between adjacent sails. The parachute canopy, including the rings and sails, is shown in more detail in Figure 1.2.

Figure 1.2: Orion CEV main parachute canopy. The rings are located near the vent and are separated by gaps. The sails start below the rings and continue to the skirt.

Like the earlier Apollo capsules, the Orion Crew Module will use a cluster of parachutes for recovery after reentry. The CEV Parachute Assembly System uses three ringsail main parachutes during the final descent stage. Each main parachute has a nominal diameter of approximately 120 ft and a quarter spherical constructed shape. The canopy has 80 gores, and each gore contains four rings and nine sails from the vent to the skirt. Each suspension line is approximately 130 ft long, and the riser is approximately 100 ft long. The capsule weight is subject to change as the capsule undergoes design variations, but the current estimate at the time of writing is approximately 19,200 lbs. During developmental testing, several two-parachute clusters will be used since this represents the worst case scenario. A two-parachute cluster is shown in Figure 1.3.

The main parachutes employ a reefing technique during deployment to permit incremental opening of the parachute and to protect the integrity of the canopy fabric.

Figure 1.3: Orion CEV two-parachute cluster.

When the parachute initially exits the deployment bag, the skirt is constricted by reefing lines to prevent the canopy from fully inflating. At preselected time intervals, the reefing lines are automatically cut and the parachute is "disreefed". Disreefing is discussed in more detail in Chapter 8.

1.3 Overview

Chapter 2 describes the governing equations for the parachute structure and surrounding fluid including the Navier–Stokes equations of incompressible flows and the structural mechanics equations of motion.

Chapter 3 describes the finite element formulations used to discretize the governing equations from Chapter 2. The structural mechanics part uses a semi-discrete formulation, and the fluid mechanics part uses a space–time formulation. The structural and fluid mechanics parts are coupled using a space–time FSI technique.

Chapter 4 explains the special techniques that have been developed specifically

for single parachute and parachute cluster computations. These include a smoothing technique to deal with incompatible structural and fluid meshes, a model for fabric and geometric porosity, a technique for projecting fluid tractions to the structure interface, and symmetric FSI.

Chapter 5 presents a FSI computation for a single parachute that analyzes the effect of design variations on parachute performance.

Chapter 6 presents FSI computations for parachute clusters that are part of an analysis to determine how the parameters representing payload models and starting conditions affect long-term cluster dynamics. A technique for modeling the contact between parachutes in a cluster is also described.

Chapter 7 presents special techniques that have been developed for fast and approximate engineering analysis of parachute dynamics. Methods are presented for parachute descent speed decomposition and model parameter extraction from parachute FSI computations.

Chapter 8 presents a single parachute disreef computation and explains the techniques for addressing specific challenges associated with reefed parachute computations. The results from disreef computations are considered preliminary.

Chapter 9 provides a summary of results and conclusions.

Chapter 2

Governing Equations

This chapter presents the governing equations for the parachute structure and surrounding fluid. The fluid mechanics of the flowfield around the parachute is governed by the Navier–Stokes equations of incompressible flows. The parachute membrane and cable deformations are governed by structural mechanics equations of motion.

2.1 Fluid Mechanics

Let $\Omega_t \subset \mathbb{R}^n_{\text{sd}}$ be the spatial domain with boundary Γ_t at time $t \in (0, T)$. The subscript t indicates the time-dependence of the domain. The Navier–Stokes equations of incompressible flows are written on Ω_t and $\forall t \in (0, T)$ as

$$\rho\left(\frac{\partial \boldsymbol{u}}{\partial t} + \boldsymbol{u} \cdot \boldsymbol{\nabla} \boldsymbol{u} - \mathbf{f}\right) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma} = \mathbf{0} \,, \tag{2.1}$$

$$\boldsymbol{\nabla} \cdot \boldsymbol{u} = 0 \,, \tag{2.2}$$

where ρ, $\boldsymbol{u}$ and $\mathbf{f}$ are the density, velocity and the external force, respectively. The stress tensor $\boldsymbol{\sigma}$ is defined as

$$\boldsymbol{\sigma}(p, \boldsymbol{u}) = -p\mathbf{I} + 2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}) \,, \tag{2.3}$$

with

$$\boldsymbol{\varepsilon}(\boldsymbol{u}) = \frac{1}{2}\left((\boldsymbol{\nabla u}) + (\boldsymbol{\nabla u})^T\right) \ . \tag{2.4}$$

Here p is the pressure, $\mathbf{I}$ is the identity tensor, $\mu = \rho\nu$ is the viscosity, ν is the kinematic viscosity, and $\boldsymbol{\varepsilon}(\boldsymbol{u})$ is the strain-rate tensor. The essential and natural boundary conditions for Eq. (2.1) are represented as

$$\boldsymbol{u} = \mathbf{g} \text{ on } (\Gamma_t)_g \ , \tag{2.5}$$

$$\mathbf{n} \cdot \boldsymbol{\sigma} = \mathbf{h} \text{ on } (\Gamma_t)_h \ , \tag{2.6}$$

where $(\Gamma_t)_g$ and $(\Gamma_t)_h$ are complementary subsets of the boundary Γ_t, $\mathbf{n}$ is the unit normal vector, and $\mathbf{g}$ and $\mathbf{h}$ are given functions. A divergence-free velocity field $\boldsymbol{u}_0(\mathbf{x})$ is specified as the initial condition.

2.2 Structural Mechanics

Let $\Omega_t^s \subset I\!R^{n_{xd}}$ be the spatial domain with boundary Γ_t^s, where $n_{xd} = 2$ for membranes and $n_{xd} = 1$ for cables. The superscript "s" indicates the structure. The parts of Γ_t^s corresponding to the essential and natural boundary conditions are represented by $(\Gamma_t^s)_g$ and $(\Gamma_t^s)_h$. The equations of motion are written as

$$\rho^s \left(\frac{d^2\mathbf{y}}{dt^2} + \eta \frac{d\mathbf{y}}{dt} - \mathbf{f}^s \right) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}^s \ = \mathbf{0} \ , \tag{2.7}$$

where ρ^s, $\mathbf{y}$, $\mathbf{f}^s$ and $\boldsymbol{\sigma}^s$ are the material density, structural displacement, external force and the Cauchy stress tensor, respectively. Here η is an artificial damping coefficient, which is nonzero only in computations where time accuracy is not required, such as in determining the deformed shape of the structure for specified fluid mechanics

forces acting on it. Such computations typically precede any fluid mechanics or FSI computations, and the artificial damping facilitates reaching that initial shape in a robust way. As such, structural dampening may be used during shape determination iterations. The stresses are expressed in terms of the second Piola–Kirchoff stress tensor $\mathbf{S}$, which is related to the Cauchy stress tensor through a kinematic transformation. Under the assumption of large displacements and rotations, small strains, and no material damping, the membranes and cables are characterized with linearly-elastic material properties. For membranes, under the assumption of plane stress, $\mathbf{S}$ becomes:

$$S^{ij} = \left(\bar{\lambda}^s G^{ij} G^{kl} + \mu^s \left(G^{il} G^{jk} + G^{ik} G^{jl}\right)\right) E_{kl} \ , \tag{2.8}$$

where for isotropic plane stress $\bar{\lambda}^s = 2\lambda^s \mu^s / (\lambda^s + 2\mu^s)$, and the Green–Lagrange strain tensor is defined as

$$E_{kl} = \frac{1}{2} \left(g_{kl} - G_{kl}\right) . \tag{2.9}$$

Here, λ^s and μ^s are the Lamé constants, G^{ij} are the contravariant components of the metric tensor in the undeformed configuration, and g_{kl} and G_{kl} are covariant metric tensor components in the deformed and undeformed configurations. For cables, under the assumption of uniaxial tension, $\mathbf{S}$ becomes $S^{11} = E_c G^{11} G^{11} E_{11}$, where E_c is the Young's modulus for the cable.

Chapter 3

Finite Element Formulations

3.1 DSD/SST Formulation of Fluid Mechanics

In the Deforming-Spatial-Domain/Stabilized Space-Time (DSD/SST) method [36, 42, 44, 38], the finite element formulation is written over a sequence of N space–time slabs Q_n, where Q_n is the slice of the space–time domain between the time levels t_n and t_{n+1}. At each time step, the integrations are performed over Q_n. The space–time finite element interpolation functions are continuous within a space–time slab, but discontinuous from one space–time slab to another. The notation $(\cdot)_n^-$ and $(\cdot)_n^+$ will denote the function values at t_n as approached from below and above. Each Q_n is decomposed into elements Q_n^e, where $e = 1, 2, \ldots, (n_{el})_n$. The subscript n used with n_{el} is for the general case where the number of space–time elements may change from one space–time slab to another. The essential and natural boundary conditions are enforced over $(P_n)_{\mathrm{g}}$ and $(P_n)_{\mathrm{h}}$, the complementary subsets of the lateral boundary of the space–time slab. The finite element trial function spaces $(\mathcal{S}_{\boldsymbol{u}}^h)_n$ for velocity and $(\mathcal{S}_p^h)_n$ for pressure, and the test function spaces $(\mathcal{V}_{\boldsymbol{u}}^h)_n$ and $(\mathcal{V}_p^h)_n = (\mathcal{S}_p^h)_n$ are defined by using, over Q_n, first-order polynomials in space and time.

The DSD/SST formulation (from [38]) is written as follows: given $(\boldsymbol{u}^h)_n^-$, find

$\boldsymbol{u}^h \in (\mathcal{S}_{\boldsymbol{u}}^h)_n$ and $p^h \in (\mathcal{S}_p^h)_n$ such that $\forall \mathbf{w}^h \in (\mathcal{V}_{\boldsymbol{u}}^h)_n$ and $\forall q^h \in (\mathcal{V}_p^h)_n$:

$$
\begin{aligned}
&\int_{Q_n} \mathbf{w}^h \cdot \rho \left(\frac{\partial \boldsymbol{u}^h}{\partial t} + \boldsymbol{u}^h \cdot \boldsymbol{\nabla} \boldsymbol{u}^h - \mathbf{f}^h \right) dQ + \int_{Q_n} \boldsymbol{\varepsilon}(\mathbf{w}^h) : \boldsymbol{\sigma}(p^h, \boldsymbol{u}^h)\, dQ \\
- &\int_{(P_n)_{\mathrm{h}}} \mathbf{w}^h \cdot \mathbf{h}^h \, dP + \int_{Q_n} q^h \boldsymbol{\nabla} \cdot \boldsymbol{u}^h \, dQ + \int_{\Omega_n} (\mathbf{w}^h)_n^+ \cdot \rho \left((\boldsymbol{u}^h)_n^+ - (\boldsymbol{u}^h)_n^- \right) d\Omega \\
+ &\sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \frac{1}{\rho} \left[\tau_{\mathrm{SUPG}} \rho \left(\frac{\partial \mathbf{w}^h}{\partial t} + \boldsymbol{u}^h \cdot \boldsymbol{\nabla} \mathbf{w}^h \right) + \tau_{\mathrm{PSPG}} \boldsymbol{\nabla} q^h \right] \cdot \left[\text{Ł}(p^h, \boldsymbol{u}^h) - \rho \mathbf{f}^h \right] dQ \\
+ &\sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \nu_{\mathrm{LSIC}} \boldsymbol{\nabla} \cdot \mathbf{w}^h \rho \boldsymbol{\nabla} \cdot \boldsymbol{u}^h \, dQ = 0,
\end{aligned}
\tag{3.1}
$$

where

$$
\text{Ł}(q^h, \mathbf{w}^h) = \rho \left(\frac{\partial \mathbf{w}^h}{\partial t} + \boldsymbol{u}^h \cdot \boldsymbol{\nabla} \mathbf{w}^h \right) - \boldsymbol{\nabla} \cdot \boldsymbol{\sigma}(q^h, \mathbf{w}^h). \tag{3.2}
$$

This formulation is applied to all space–time slabs $Q_0, Q_1, Q_2, \ldots, Q_{N-1}$, starting with $(\boldsymbol{u}^h)_0^- = \boldsymbol{u}_0$. Here τ_{SUPG}, τ_{PSPG} and ν_{LSIC} are the SUPG (Streamline-Upwind/Petrov-Galerkin), PSPG (Pressure-Stabilizing/Petrov-Galerkin) and LSIC (least-squares on incompressibility constraint) stabilization parameters. There are various ways of defining these stabilization parameters. Here we provide the definitions given in [38]:

$$
\tau_{\mathrm{SUPG}} = \left(\frac{1}{\tau_{\mathrm{SUGN12}}^2} + \frac{1}{\tau_{\mathrm{SUGN3}}^2} \right)^{-\frac{1}{2}}, \tag{3.3}
$$

$$
\tau_{\mathrm{SUGN12}} = \left(\sum_{a=1}^{n_{en}} \left| \frac{\partial N_a}{\partial t} + \boldsymbol{u}^h \cdot \boldsymbol{\nabla} N_a \right| \right)^{-1}, \tag{3.4}
$$

$$
\tau_{\mathrm{SUGN3}} = \frac{h_{\mathrm{RGN}}^2}{4\nu}, \tag{3.5}
$$

$$
h_{\mathrm{RGN}} = 2 \left(\sum_{a=1}^{n_{en}} |\mathbf{r} \cdot \boldsymbol{\nabla} N_a| \right)^{-1}, \tag{3.6}
$$

$$
\mathbf{r} = \frac{\boldsymbol{\nabla} \| \boldsymbol{u}^h \|}{\| \boldsymbol{\nabla} \| \boldsymbol{u}^h \| \|}, \tag{3.7}
$$

$$
\tau_{\mathrm{PSPG}} = \tau_{\mathrm{SUPG}}, \tag{3.8}
$$

and in [46]:

$$\nu_{\text{LSIC}} = \tau_{\text{SUPG}} \left\| \boldsymbol{u}^h - \mathbf{v}^h \right\|^2, \tag{3.9}$$

where n_{en} is the number of (space–time) element nodes, N_a is the space–time shape function associated with the space–time node a, and $\mathbf{v}^h$ is the mesh velocity. As an alternative to the construction of τ_{SUPG} as given by Eqs. (3.3)–(3.4), another option was introduced in [46]. In that option, τ_{SUPG} is constructed based on separate definitions for the advection-dominated and transient-dominated limits:

$$\tau_{\text{SUPG}} = \left(\frac{1}{\tau_{\text{SUGN1}}^2} + \frac{1}{\tau_{\text{SUGN2}}^2} + \frac{1}{\tau_{\text{SUGN3}}^2} \right)^{-\frac{1}{2}}, \tag{3.10}$$

$$\tau_{\text{SUGN1}} = \left(\sum_{a=1}^{n_{en}} \left| \left(\boldsymbol{u}^h - \mathbf{v}^h \right) \cdot \boldsymbol{\nabla} N_a \right| \right)^{-1}, \tag{3.11}$$

$$\tau_{\text{SUGN2}} = \frac{\Delta t}{2}, \tag{3.12}$$

where Δt is the time-step size. It was noted in [46] that separating τ_{SUGN12} into its advection- and transient-dominated components as given by Eqs. (3.11)–(3.12) is equivalent to excluding the $\left(\frac{\partial N_a}{\partial t}\big|_{\xi} \right)$ part of $\left(\frac{\partial N_a}{\partial t} \right)$ in Eq. (3.4), making that the definition for τ_{SUGN1}, and accounting for $\left(\frac{\partial N_a}{\partial t}\big|_{\xi} \right)$ in the definition for τ_{SUGN2} given by Eq. (3.12). Here ξ is the vector of element coordinates. For more ways of calculating τ_{SUPG}, τ_{PSPG} and ν_{LSIC}, see [45, 1, 38, 39, 2, 41, 24, 4, 5, 10, 7]. References [38, 39, 41] also include the Discontinuity-Capturing Directional Dissipation (DCDD) stabilization, which was introduced as an alternative to the LSIC stabilization.

Several of the remarks from [54] and [46] concerning this chapter are relevant and are presented here as Remarks 1–8.

Remark 1 *As an alternative to how the SUPG test function is defined in Eq. (3.1), another option was proposed in [46]. In this option, the SUPG test function* $\left(\frac{\partial \mathbf{w}^h}{\partial t} + \boldsymbol{u}^h \cdot \boldsymbol{\nabla} \mathbf{w}^h \right)$ *is replaced with* $\left(\left(\boldsymbol{u}^h - \mathbf{v}^h \right) \cdot \boldsymbol{\nabla} \mathbf{w}^h \right)$. *This replacement is equivalent to excluding the*

$\left(\left.\frac{\partial \mathbf{w}^h}{\partial t}\right|_\xi\right)$ *part of* $\left(\frac{\partial \mathbf{w}^h}{\partial t}\right)$. *In [46], this option was called "WTSE", and the option where the* $\left(\left.\frac{\partial \mathbf{w}^h}{\partial t}\right|_\xi\right)$ *term is active, "WTSA".*

Remark 2 *The stability and accuracy analysis reported in [34] for the DSD/SST formulation of the time-dependent advection equation shows for linear functions in space and time that the WTSA option yields higher-order accuracy than the WTSE option.*

Remark 3 *With the function spaces defined in the paragraph preceding Eq. (3.1), for each space–time slab velocity and pressure assume double unknown values at each spatial node. One value corresponds to the lower end of the slab, and the other one the upper end. In [46], the option of using double unknown values at a spatial node is called "DV" for velocity and "DP" for pressure. In this case, as pointed out in [46], we use two integration points over the time interval of the space–time slab, and this time-integration option is called "TIP2". This version of the DSD/SST formulation, with the options set DV, DP and TIP2, is called "DSD/SST-DP".*

Remark 4 *In [46], the option of using, for each space–time slab, a single unknown pressure value at each spatial node was proposed with the option name "SP". With this, another version of the DSD/SST formulation was proposed in [46], where the options set is DV, SP and TIP2. This version is called "DSD/SST-SP". Because the number of unknown pressure values is halved, the computational cost is reduced substantially.*

Remark 5 *To reduce the computational cost further, the option of using only one integration point over the time interval of the space–time slab was proposed in [46]. This time-integration option is called "TIP1". With that, a third version of the DSD/SST formulation was proposed in [46], where the options set is DV, SP and TIP1. This version is called "DSD/SST-TIP1".*

Remark 6 *For DSD/SST-SP and DSD/SST-TIP1, in integration of the incompressibility-constraint term over each space–time slab, as proposed in [31], we use only one integration point in time, shifted to the upper time level of the slab. All other terms in the space–time finite element formulation are integrated by using Gaussian quadrature points in time, with the number of points set to whatever we intended to have for the overall formulation. With this technique, as pointed out in [31], the incompressibility constraint equation focuses on the velocity field* $(\boldsymbol{u}^h)^-_{n+1}$.

3.2 Semi-Discrete Formulation of Structural Mechanics

With $\mathbf{y}^h$ and $\mathbf{w}^h$ coming from appropriately defined trial and test function spaces, respectively, the semi-discrete finite element formulation of the structural mechanics equations (see [17, 3, 28]) is written as

$$\int_{\Omega^s_0} \mathbf{w}^h \cdot \rho^s \frac{d^2\mathbf{y}^h}{dt^2}\, d\Omega^s + \int_{\Omega^s_0} \mathbf{w}^h \cdot \eta\rho^s \frac{d\mathbf{y}^h}{dt}\, d\Omega^s + \int_{\Omega^s_0} \delta\mathbf{E}^h : \mathbf{S}^h\, d\Omega^s = \int_{\Omega^s_t} \mathbf{w}^h \cdot \left(\mathbf{t}^h + \rho^s \mathbf{f}^s\right) d\Omega^s. \tag{3.13}$$

The fluid mechanics forces acting on the structure are represented by vector $\mathbf{t}^h$. The left-hand-side terms of Eq. (3.13) are referred to in the original configuration and the right-hand-side terms in the deformed configuration at time t. From this formulation at each time step we obtain a nonlinear system of equations. In solving that nonlinear system with an iterative method, we use an incremental form (see [17, 3, 28, 14]), which is expressed as

$$\left[\frac{\mathbf{M}}{\beta\Delta t^2} + \frac{(1-\alpha)\,\gamma\mathbf{C}}{\beta\Delta t} + (1-\alpha)\,\mathbf{K}\right]\Delta\mathbf{d}^i = \mathbf{R}^i. \tag{3.14}$$

Here $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ is the damping matrix, $\mathbf{K}$ is the consistent tangent matrix associated with the internal elastic forces, $\mathbf{R}^i$ is the residual vector at the i^{th} iteration, and $\Delta\mathbf{d}^i$ is the i^{th} increment in the nodal displacements vector $\mathbf{d}$. For spatially-constant η, the damping matrix can be written as $\mathbf{C} = \eta\mathbf{M}$. All of the terms known from the previous iteration are lumped into the residual vector $\mathbf{R}^i$. The parameters α, β, γ are part of the Hilber–Hughes–Taylor [9] scheme, which is the time-integration technique used here. For all computations reported here, the structural mechanics part of the mass matrix is lumped. This is consistent with other parachute computations performed by the T⋆AFSM.

3.3 Stabilized Space–Time Fluid–Structure Interaction (SSTFSI) Technique

The SSTFSI technique was introduced in [46], where it was described based on the finite element formulations given by Eqs. (3.1) and (3.13), with a slight change of notation and with a clarification of how the fluid–structure interface conditions are handled. In that notation subscripts 1 and 2 refer to fluid and structure, respectively. Furthermore, while subscript I refers to the fluid–structure interface, subscript E refers to "elsewhere" in the fluid and structure domains or boundaries. Here we write from [46] the equations representing the SSTFSI technique:

$$
\begin{aligned}
&\int_{Q_n} \mathbf{w}_{1\mathrm{E}}^h \cdot \rho \left(\frac{\partial \boldsymbol{u}^h}{\partial t} + \boldsymbol{u}^h \cdot \boldsymbol{\nabla} \boldsymbol{u}^h - \mathbf{f}^h \right) dQ + \int_{Q_n} \boldsymbol{\varepsilon}(\mathbf{w}_{1\mathrm{E}}^h) : \boldsymbol{\sigma}(p^h, \boldsymbol{u}^h)\, dQ \\
- &\int_{(P_n)_{\mathrm{h}}} \mathbf{w}_{1\mathrm{E}}^h \cdot \mathbf{h}_{1\mathrm{E}}^h\, dP + \int_{Q_n} q_{1\mathrm{E}}^h \boldsymbol{\nabla} \cdot \boldsymbol{u}^h\, dQ + \int_{\Omega_n} (\mathbf{w}_{1\mathrm{E}}^h)_n^+ \cdot \rho \left((\boldsymbol{u}^h)_n^+ - (\boldsymbol{u}^h)_n^- \right) d\Omega \\
+ &\sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \frac{1}{\rho} \left[\tau_{\mathrm{SUPG}} \rho \left(\frac{\partial \mathbf{w}_{1\mathrm{E}}^h}{\partial t} + \boldsymbol{u}^h \cdot \boldsymbol{\nabla} \mathbf{w}_{1\mathrm{E}}^h \right) + \tau_{\mathrm{PSPG}} \boldsymbol{\nabla} q_{1\mathrm{E}}^h \right] \cdot \left[\mathrm{L}(p^h, \boldsymbol{u}^h) - \rho \mathbf{f}^h \right] dQ \\
+ &\sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \nu_{\mathrm{LSIC}} \boldsymbol{\nabla} \cdot \mathbf{w}_{1\mathrm{E}}^h \rho \boldsymbol{\nabla} \cdot \boldsymbol{u}^h\, dQ = 0,
\end{aligned}
\tag{3.15}
$$

$$\int_{Q_n} q_{1\mathrm{I}}^h \boldsymbol{\nabla}\cdot\boldsymbol{u}^h \, dQ + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} \frac{1}{\rho}\left[\tau_{\mathrm{PSPG}}\boldsymbol{\nabla} q_{1\mathrm{I}}^h\right]\cdot\left[\mathrm{L}(p^h,\boldsymbol{u}^h) - \rho\mathbf{f}^h\right] dQ = 0, \tag{3.16}$$

$$\int_{(\Gamma_{1\mathrm{I}})_{\mathrm{REF}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}\cdot\left((\mathbf{x}_{1\mathrm{I}}^h)_{n+1} - (\mathbf{x}_{2\mathrm{I}}^h)_{n+1}\right) d\Gamma = 0, \tag{3.17}$$

$$\int_{(\Gamma_{1\mathrm{I}})_{\mathrm{REF}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^{-}\cdot\left((\boldsymbol{u}_{1\mathrm{I}}^h)_{n+1}^{-} - \boldsymbol{u}_{2\mathrm{I}}^h\right) d\Gamma = 0\,, \tag{3.18}$$

$$\begin{aligned}\int_{(P_n)_{\mathrm{h}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^{-}\cdot\mathbf{h}_{1\mathrm{I}}^h \, dP = & - \int_{(P_n)_{\mathrm{h}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^{-}\cdot p\mathbf{n}\, dP + \int_{Q_n} 2\mu\boldsymbol{\varepsilon}((\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^{-}) : \boldsymbol{\varepsilon}(\boldsymbol{u})\, dQ \\ & + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^{-}\cdot\boldsymbol{\nabla}\cdot(2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}))\; dQ,\end{aligned} \tag{3.19}$$

$$\int_{(\Omega_{2\mathrm{I}})_{\mathrm{REF}}} \mathbf{w}_{2\mathrm{I}}^h\cdot\left(\mathbf{h}_{2\mathrm{I}}^h + \left(\mathbf{h}_{1\mathrm{I}}^h\right)_{\mathrm{A}} + \left(\mathbf{h}_{1\mathrm{I}}^h\right)_{\mathrm{B}}\right) d\Omega = 0, \tag{3.20}$$

$$\begin{aligned}&\int_{(\Omega_2)_0} \mathbf{w}_2^h\cdot\rho_2\frac{d^2\mathbf{y}^h}{dt^2}\, d\Omega + \int_{(\Omega_2)_0} \mathbf{w}_2^h\cdot\eta\rho_2\frac{d\mathbf{y}^h}{dt}\, d\Omega + \int_{(\Omega_2)_0} \delta\mathbf{E}^h : \mathbf{S}^h\, d\Omega \\ = & \int_{\Omega_2} \mathbf{w}_2^h\cdot\rho_2\mathbf{f}_2^h\, d\Omega + \int_{\Omega_{2\mathrm{E}}} \mathbf{w}_{2\mathrm{E}}^h\cdot\mathbf{h}_{2\mathrm{E}}^h\, d\Omega + \int_{\Omega_{2\mathrm{I}}} \mathbf{w}_{2\mathrm{I}}^h\cdot\mathbf{h}_{2\mathrm{I}}^h\, d\Omega.\end{aligned} \tag{3.21}$$

Here $(\Gamma_{1\mathrm{I}})_{\mathrm{REF}}$ and $(\Omega_{2\mathrm{I}})_{\mathrm{REF}}$ represent some reference configurations of $\Gamma_{1\mathrm{I}}$ and $\Omega_{2\mathrm{I}}$, respectively, and $\mathbf{x}_{1\mathrm{I}}^h$ and $\mathbf{x}_{2\mathrm{I}}^h$ are the fluid mechanics and structural mechanics nodal positions at the fluid–structure interface. In reconciling the slightly modified notation used here with the notation used in Eqs. (3.1) and (3.13), we note that $\rho_2 = \rho^s$, $\mathbf{f}_2^h = \mathbf{f}^s$, $(\Omega_2)_0 = \Omega_0^s$, $\Omega_2 = \Omega_t^s$, and $\Omega_{2\mathrm{I}}$ and $\Omega_{2\mathrm{E}}$ denote the partitions of Ω_2 corresponding to the interface and elsewhere. We also note that $\mathbf{h}_{2\mathrm{I}}^h = \mathbf{t}^h$, and $\left(\mathbf{h}_{1\mathrm{I}}^h\right)_{\mathrm{A}}$ and $\left(\mathbf{h}_{1\mathrm{I}}^h\right)_{\mathrm{B}}$

represent the values of $\mathbf{h}_{1\mathrm{I}}^h$ associated with the fluid surfaces above and below the membrane structure. The symbol $\mathbf{h}_{2\mathrm{E}}^h$ denotes the prescribed external forces acting on the structure in $\Omega_{2\mathrm{E}}$, which is separate from $\mathbf{f}_2^h$. In this formulation, $(\boldsymbol{u}_{1\mathrm{I}}^h)_{n+1}^-$, $\mathbf{h}_{1\mathrm{I}}^h$ and $\mathbf{h}_{2\mathrm{I}}^h$ (the fluid velocity, fluid stress and structural stress at the fluid–structure interface) are treated as separate unknowns, and Eqs. (3.18), (3.19) and (3.20) can be seen as corresponding to these three unknowns, respectively. The structural displacement rate at the interface, $\boldsymbol{u}_{2\mathrm{I}}^h$, is derived from $\mathbf{y}^h$.

We note that Eq. (3.19) has been derived by assuming that the viscous-flux jump terms across inter-element borders are negligible. We also note that the last term of that equation, in its original form in [46], was written as a global integral $\int_{Q_n}$ rather than a series of element-level integrals. Alternatively, one can leave that projection equation in its form prior to the integration-by-parts:

$$\begin{aligned}\int_{(P_n)_{\mathrm{h}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \cdot \mathbf{h}_{1\mathrm{I}}^h \, dP = \; & - \int_{(P_n)_{\mathrm{h}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \cdot p\mathbf{n} \, dP \\ & + \int_{(P_n)_{\mathrm{h}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \cdot (\mathbf{n} \cdot (2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}))) \, dP, \qquad (3.22)\end{aligned}$$

and this would require also the projection of $\boldsymbol{\varepsilon}(\boldsymbol{u})$ from the element interiors to the nodes.

The formulation given by Eqs. (3.15)–(3.21) is based on allowing for cases when the fluid and structure meshes at the interface are not identical. If they are identical, as pointed out in [46], the same formulation can still be used.

It was noted in [46] that, for constant viscosity, the term $\boldsymbol{\nabla} \cdot (2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}))$ in Eq. (3.19) vanishes for tetrahedral elements. It was further noted in [46] that the same statement can be made also in the context of that term being a part of the expression $\mathrm{L}(p^h, \boldsymbol{u}^h)$ appearing in Eqs. (3.15) and (3.16).

Remark 7 *In FSI computations with membranes and shells, the pressure at the interface has split nodal values corresponding to the fluid surfaces above and below the*

membrane or shell structure. We propose to use such split nodal values for pressure also at the boundaries (i.e. edges) of a membrane structure surrounded by the fluid. Our computations show that this provides additional numerical stability for the edges of the membrane.

Remark 8 *In [46], the version of the SSTFSI technique corresponding to the DSD/SST-TIP1 formulation was called "SSTFSI-TIP1".*

3.4 Mesh Update Methods

The mesh update methods, and related techniques, developed by the T★AFSM to be used in conjunction with the DSD/SST formulation was described in Section 4 of [46]. The section includes references to a number of articles where these methods and techniques were described in detail (see [36, 43, 35, 12, 13, 29, 30]), as well as references to related methods developed by other researchers (see [8]).

3.5 Solution of the Fully-Discretized Coupled Equations

Full discretization of the FSI formulation described in Section 3.3 leads to coupled, nonlinear equation systems that need to be solved at every time step. The techniques developed by the T★AFSM for that purpose were described in Section 5 of [46]. The section includes references to a number of articles where related techniques were described in detail (see [37, 39, 47, 48, 53, 40]), as well as references to related methods developed by other researchers (see [11]).

Chapter 4

Special Techniques

This chapter describes several special techniques that have been developed by the T★AFSM to address specific computational challenges related to parachute FSI modeling.

4.1 Directional Geometric Smoothing Technique

As mentioned in Chapter 1, ringsail parachutes are constructed with hundreds of gaps and slits, and reinforcement cables that are embedded longitudinally in the canopy structure cause the formation of "peaks" and "valleys" when the canopy is inflated. Representing the geometric complexity of the structure at the interface would require a fluid mechanics mesh that is not computationally affordable. To make the problem more tractable, we use a structural interface that represents the actual parachute geometry and a coarser smoothened fluid interface to approximate the geometry using fewer nodes and a simpler surface shape. Certain nodes are picked from the structure interface mesh to generate the set of fluid interface nodes, so the number of fluid interface nodes are smaller than the number of structure interface nodes. While generating the set of fluid interface nodes, the structure interface nodes from the valleys are picked. In picking these nodes circumferentially, a few valleys

can be skipped, and in picking them longitudinally, inside a valley a few nodes can be skipped. The nodes are then connected with three-node triangular elements, resulting in a smooth fluid mesh along the circumferential direction. Figure 4.1 shows the structural and fluid interface meshes.

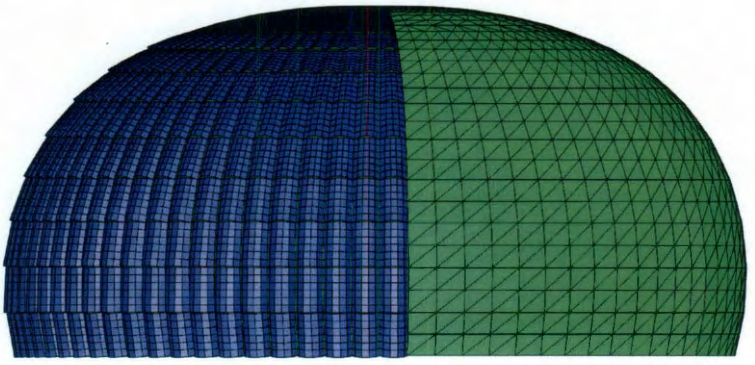


Figure 4.1: Structural mechanics mesh (left) and fluid interface mesh (right) for a single parachute.

Because the fluid and structure meshes are incompatible at the interface, nodal data cannot be transferred directly. Rather, the structural mesh and displacement rates are projected to the fluid mesh after a geometric smoothing. To address the geometric complexities of the peaks and valleys, smoothing is done in the circumferential direction of the parachute canopy. This was introduced in [51, 52] as the FSI Directional Geometric Smoothing Technique (FSI-DGST). By generating the fluid interface mesh using nodes picked circumferentially from the structure interface valleys, the preferred smoothing directions can approximately be represented by the gridlines of the interface mesh. In the FSI-DGST, a value (mesh coordinate or displacement rate) at a given node is replaced by a weighted average of the values at that node and a limited set of nearby nodes. When projecting the stress values from the smoothened interface to the structure, the values for the mapping nodes are transferred directly, and for the remaining nodes a weighted average is used.

4.2 Separated Stress Projection

An additional stress projection option called Separated Stress Projection (SSP) was proposed in [50]. In SSP, the pressure and viscous parts of the stress at the fluid interface are projected to the structure interface separately, pressure as a scalar and viscous stress as a vector. The projected parts are then combined while integrating the interface stresses in the structural mechanics equations. In the SSP option, the projections given by Eq. (3.19) and (3.20) are replaced with the following projections:

$$\int_{(P_n)_{\mathrm{h}}} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \cdot \left(\mathbf{h}_v^h\right)_{1\mathrm{I}} \, dP = \int_{Q_n} 2\mu\boldsymbol{\varepsilon}((\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^-) : \boldsymbol{\varepsilon}(\boldsymbol{u}) \, dQ + \sum_{e=1}^{(n_{el})_n} \int_{Q_n^e} (\mathbf{w}_{1\mathrm{I}}^h)_{n+1}^- \cdot \boldsymbol{\nabla} \cdot (2\mu\boldsymbol{\varepsilon}(\boldsymbol{u})) \, dQ, \tag{4.1}$$

$$\int_{(\Omega_{2\mathrm{I}})_{\mathrm{REF}}} q_{2\mathrm{I}}^h \left(p_{2\mathrm{I}}^h + \left(p_{1\mathrm{I}}^h\right)_{\mathrm{A}} - \left(p_{1\mathrm{I}}^h\right)_{\mathrm{B}}\right) d\Omega = 0, \tag{4.2}$$

$$\int_{(\Omega_{2\mathrm{I}})_{\mathrm{REF}}} \mathbf{w}_{2\mathrm{I}}^h \cdot \left(\left(\mathbf{h}_v^h\right)_{2\mathrm{I}} + \left(\left(\mathbf{h}_v^h\right)_{1\mathrm{I}}\right)_{\mathrm{A}} + \left(\left(\mathbf{h}_v^h\right)_{1\mathrm{I}}\right)_{\mathrm{B}}\right) d\Omega = 0 \, , \tag{4.3}$$

$$\mathbf{h}_{2\mathrm{I}}^h = -p_{2\mathrm{I}}^h \mathbf{n}_{2\mathrm{I}} + \left(\mathbf{h}_v^h\right)_{2\mathrm{I}} , \tag{4.4}$$

where $\mathbf{h}_v^h$ is the viscous part of the stress vector, $p_{1\mathrm{I}}^h$ is the pressure at the fluid interface, $p_{2\mathrm{I}}^h$ is the projection of that to the structure interface, and $\mathbf{n}_{2\mathrm{I}}$ is the unit normal vector at the structure interface. The stress vector at the structure interface, given by Eq. (4.4), is evaluated while integrating the interface stresses in the structural mechanics equations. Therefore, in the way Eq. (4.4) is used, $\mathbf{n}_{2\mathrm{I}}$ is evaluated at the integration point, and $p_{2\mathrm{I}}^h$ and $\left(\mathbf{h}_v^h\right)_{2\mathrm{I}}$ are the interpolated values at the integration point.

As an alternative to the projection given by Eq. (4.1), that projection equation

can be left in its form prior to the integration-by-parts:

$$\int_{(P_n)_h} (\mathbf{w}_{1I}^h)_{n+1}^- \cdot \left(\mathbf{h}_v^h\right)_{1I} \, dP = \int_{(P_n)_h} (\mathbf{w}_{1I}^h)_{n+1}^- \cdot (\mathbf{n} \cdot (2\mu\boldsymbol{\varepsilon}(\boldsymbol{u}))) \, dP, \tag{4.5}$$

which would correspond to Eq. (3.22) and would again require the projection of $\boldsymbol{\varepsilon}(\boldsymbol{u})$ from the element interiors to the nodes. Figure 4.2 provides an illustration the SSP technique.

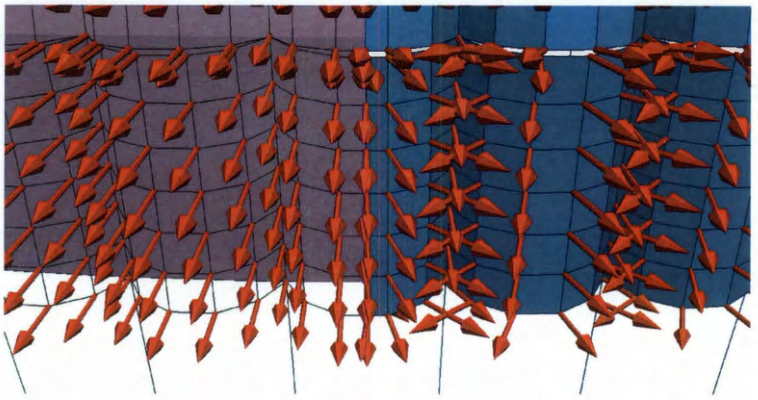


Figure 4.2: An illustration of the Separated Stress Projection (SSP) technique. The purple surface on the left half of the canopy is the fluid interface and the blue surface on the right is the structure interface. The arrows on the left represent the structure interface stress vectors obtained by transferring the total fluid interface stress vectors directly to the structure. The arrows on the right represent the structure interface stress vectors obtained with the SSP technique.

4.3 Homogenized Modeling of Geometric Porosity

The total porosity of a ringsail parachute is a combination of fabric permeability and hundreds of gaps and slits inherent in the canopy design. To simulate air flow through the canopy, a fluid velocity, crossing from the high-pressure side to the low-pressure side, is generated at each node of the fluid interface. This velocity is proportional to the cross-canopy pressure differential. The fabric porosity is circumferentially-uniform and varies from the vent to the skirt. Fabric porosity coefficients for each

region of the canopy are known and typically reported in units of "CFM," meaning "cubic feet of air per minute per square foot at a pressure differential of half an inch of water". For example, when a fabric with a porosity coefficient of 1 CFM is subjected to a pressure differential of $\frac{1}{2}$ in of H_2O, a volumetric flowrate of 1 ft^3/min occurs across a unit surface area of 1 ft^2, which translates to a normal velocity of 1 ft/min. As explained in Section 4.1, the fluid interface mesh generated using FSI-DGST is a smooth representation that does not include the geometric complexities of the canopy. Therefore, the geometric porosity must be accounted for using a special technique.

4.3.1 HMGP Technique

Homogenized Modeling of Geometric Porosity (HMGP) was introduced in [50, 49] to model the combination of fabric and geometric porosity as an equivalent, locally-varying fabric porosity. We divide the canopy into concentric patches and calculate an equivalent fabric-porosity coefficient for each. Each patch includes a slit and part of a ring or sail on either side of the slit. A porosity coefficient is calculated for each patch, and at the border between two patches the average of the two porosity coefficients is used. To calculate the porosity coefficient for each patch, we carry out a one-time flow computation, holding the canopy rigid and using a thin slice of the canopy with a small number of gores, with all the rings, sails and slits. Using only a four-gore slice, as shown in Figure 4.3, keeps the problem size at a manageable level. The porosity coefficient for a patch J can then be calculated using the following expression:

$$\frac{\dot{V}_J}{(A_1)_J} = -(k_{PORO})_J \frac{\Delta F_J}{(A_2)_J}. \tag{4.6}$$

The area of the patch J calculated using the smoothened fluid interface is denoted by $(A_1)_J$, and the area calculated using the structure interface is denoted by $(A_2)_J$. With the additional notation of $(A_F)_J$ representing the fabric area and $(A_G)_J$ representing

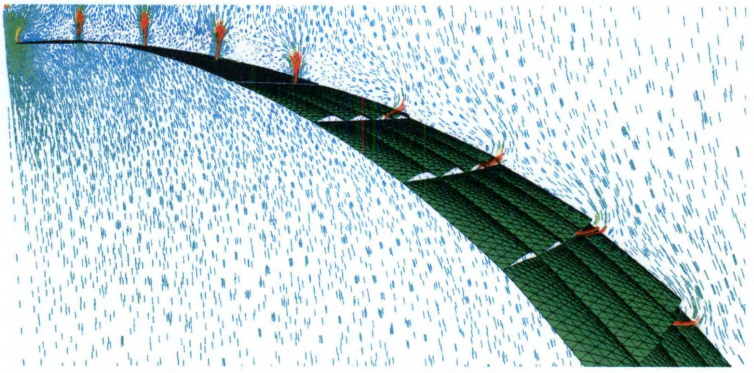


Figure 4.3: Four-gore fluid mechanics computation to calculate porosity coefficients.

the gap (or slit) area, we can write $(A_2)_J = (A_F)_J$. The symbol $\dot{V}_J$ represents the volumetric flow rate crossing the patch J. It is the sum of the flow passing through the gap (or slit) and the flow through the fabric due to its porosity:

$$\dot{V}_J = \left(\dot{V}_F\right)_J + \left(\dot{V}_G\right)_J, \tag{4.7}$$

where $\left(\dot{V}_F\right)_J$ and $\left(\dot{V}_G\right)_J$ are calculated by integrating the flow over $(A_F)_J$ and $(A_G)_J$, respectively. The pressure differential seen when crossing the patch J is integrated over its area to yield a force differential denoted by ΔF_J:

$$\Delta F_J = \int_{(A_2)_J} \Delta p \, dA. \tag{4.8}$$

We define the average pressure differential across the structure for the patch J as

$$\Delta p_J = \frac{\Delta F_J}{(A_2)_J}, \tag{4.9}$$

and from Eq. (4.6) rewrite the expression used for calculating the homogenized porosity as

$$\frac{\dot{V}_J}{(A_1)_J} = -\left(k_{PORO}\right)_J \Delta p_J. \tag{4.10}$$

In computations that use HMGP, Eq. (3.18) is replaced with the following one:

$$\int_{\Gamma_{1I}} (\mathbf{w}_{1I}^h)_{n+1}^- \cdot \left((\boldsymbol{u}_{1I}^h)_{n+1}^- - \boldsymbol{u}_{2I}^h + k_{\text{PORO}} \left(\mathbf{n} \cdot \mathbf{h}_{1I}^h \right) \mathbf{n} \right) d\Gamma = 0 \quad , \tag{4.11}$$

In the current implementation of Eq. (4.15), only the pressure component of $\mathbf{h}_{1I}^h$ is taken into account.

4.3.2 HMGP-FG Technique

An improvement to HMGP was introduced in [32]. This new technique, which is called HMGP-FG, separates the fabric and geometric porosity contributions. The velocity coming from the fabric contribution varies linearly with pressure differential, and the velocity coming from the geometric contribution varies nonlinearly.

In HMGP-FG, instead of using a single expression for $\dot{V}_J$ as given by Eq. (4.10), we use separate expressions for $\left(\dot{V}_F\right)_J$ and $\left(\dot{V}_G\right)_J$, with separate porosity coefficients $(k_F)_J$ and $(k_G)_J$ as follows:

$$\frac{\left(\dot{V}_F\right)_J}{(A_1)_J} = -(k_F)_J \frac{(A_F)_J}{(A_1)_J} \Delta p_J, \tag{4.12}$$

$$\frac{\left(\dot{V}_G\right)_J}{(A_1)_J} = -(k_G)_J \frac{(A_G)_J}{(A_1)_J} sgn(\Delta p_J) \sqrt{\frac{|\Delta p_J|}{\rho}}, \tag{4.13}$$

where ρ is the fluid density. Then, the normal velocity crossing the fluid interface is modeled nodally using the following expression:

$$u_n = -(k_F)_J \frac{A_F}{A_1} \Delta p - (k_G)_J \frac{A_G}{A_1} sgn(\Delta p) \sqrt{\frac{|\Delta p|}{\rho}}, \tag{4.14}$$

where $(k_F)_J$, $(k_G)_J$, A_F, A_G and A_1 can be seen as "material properties", calculated for each node by area-weighted averaging of the "material properties" of the (triangular) fluid interface elements sharing that node. Each fluid interface element belongs

to a "material properties" group. Each structural interface (fabric) element and each gap (or slit) also belongs to a "material properties" group. Each group is associated with a patch J. The values of $(k_F)_J$ and $(k_G)_J$ for a group come from the patch J that the group is associated with. The symbols A_F, A_G and A_1 represent for a group the total instantaneous area of the fabric, the sum of the instantaneous areas of the gap(s) and the sum of the areas of the fluid interface elements. In this new version of the HMGP we have 14 patches, with no gaps or slits in the first and last patches, and the groups are defined based on these 14 patches. Longitudinally, each group spans over one patch. Circumferentially, each group spans over 4 gores in Patch 1, 2 gores in Patches 2–5, and 1 gore in Patches 6–14.

In computations that use HMGP-FG, Eq. (3.18) is replaced with the following one:

$$\int_{\Gamma_{1I}} (\mathbf{w}_{1I}^h)_{n+1}^- \cdot \left((\boldsymbol{u}_{1I}^h)_{n+1}^- - \boldsymbol{u}_{2I}^h + k_F \left(\mathbf{n} \cdot \mathbf{h}_{1I}^h\right) \mathbf{n} + k_G \sqrt{\left(\mathbf{n} \cdot \mathbf{h}_{1I}^h\right) \mathbf{n}} \right) d\Gamma = 0 \ , \quad (4.15)$$

In the current implementation of Eq. (4.15), only the pressure component of $\mathbf{h}_{1I}^h$ is taken into account.

4.4 Symmetric FSI

In a technique called symmetric FSI, we project to the structure the circumferentially-averaged fluid interface stress, $\left(\mathbf{h}_{1I}^h\right)_{\text{AVE}}$, which is symmetric with respect to the parachute axis (see Section 3.3 for interface notation). We use symmetric FSI to build a good starting point for the full FSI computation without generating any asymmetric parachute deformation or gliding in the process. After a period of symmetric computation, we project to the structure $(1 - r_S)\, \mathbf{h}_{1I}^h + r_S \left(\mathbf{h}_{1I}^h\right)_{\text{AVE}}$, where r_S is gradually varied from 1.0 to 0.0. In the computations reported here, for expedited implementation, the symmetrization of the interface stress projected to the structure

and the de-symmetrization with the parameter r_S are done in terms of only the pressure component of the interface stress, $-p_{1I}^h \mathbf{n}$. This expedited implementation was motivated by the SSP. Before starting full FSI, the de-symmetrization is accomplished over a period of approximately 7 s by varying r_S from 1.0 to 0.0 in a Cosine form.

Chapter 5

Single Parachute Computations

The Orion capsule is expected to use a cluster of three parachutes. However, parametric studies for a single parachute can be useful for performance analysis and comparison of many design variations. The single-parachute computation presented in this chapter will also serve as an introduction to the parachute FSI modeling techniques used by the T★AFSM. Many of these techniques are used for the parachute cluster computations in Chapter 6.

Here we present a computational analysis of design variations intended to improve parachute performance. The "steady descent" (i.e. without events such as disreefing) of a ringsail parachute is dynamic, and the payload descent speed is oscillatory in nature. These oscillations are a concern for parachute design engineers because mission requirements dictate a maximum allowable descent speed. One suggested method for limiting descent speed oscillations is to restrict the periodic "breathing" motion with an Over Inflation Control Line (OICL).

Canopy loading is another concern and is defined as W_p/S_o, where W_p is the payload weight and S_o is the nominal area with a constant value of approximately $10,500$ ft^2. Increased canopy loading could result from capsule design modifications or contingencies such as the loss of a parachute during descent. Changes in canopy

loading can significantly alter the parachute aerodynamics and affect performance. Here we analyze a parachute with a 76 ft OICL and $W_p/S_o = 0.650$.

5.1 Starting Conditions

It is essential to have a good starting condition for any FSI computation, and it is especially important in parachute computations where the structure is light and very sensitive to the fluid dynamics forces. The starting condition for parachutes is defined in terms of the parachute shape, descent speed and a developed flow field. The starting parachute shape is determined by a stand-alone structural mechanics computation with a uniform parachute inflation pressure equal to the stagnation pressure corresponding to the starting descent speed. A fluid mechanics mesh is generated around the settled starting shape. The developed flow field is obtained with a two-part stand-alone fluid mechanics computation based on the starting descent speed. After obtaining a starting shape and developed flow field, we begin a symmetric FSI computation (see Section 4.4). Modifications to the payload weight and OICL length are made during symmetric FSI. At the start of that stage, the skirt diameter is 78.9 ft. We begin with a zero-stiffness OICL to allow the breathing motion to begin and turn on the stiffness when the diameter is at a minimum to avoid the sudden changes of the internal force balance for the structure. Symmetric FSI is computed for 100 s allowing the solution to settle. We note that the payload and the parachute have no horizontal speed at the end of the symmetric FSI step, which does not match what is observed in NASA drop tests. To emulate the swinging motion observed in the drop tests, we instantaneously hike the horizontal speed of the payload to 20 ft/s. Simultaneously, we begin the de-symmetrization using a Cosine form which lasts for one breathing period (7 s). Although the vortex shedding pattern behind the parachute is not exactly the same in each computation, the momentum added to the

payload with this velocity hike is consistent. We believe that emulating the payload momentum in this fashion represents this aspect of the starting condition with a reasonable closeness to the actual conditions. We use the parachute shape and flow field at the end of symmetric FSI as the starting condition for the FSI computation.

5.2 Computational Conditions

The single ringsail parachute in this computation is used with a 5,600 lb payload. At the time of computation, this represented 1/3 of the expected weight of the capsule intended to be recovered using a cluster of three parachutes. The starting descent speed is 25 ft/s, which is the estimated descent speed for the given canopy loading. The parachute geometry is described in detail in Chapter 1.

The structural mechanics mesh has 30,722 nodes, 26,000 four-node quadrilateral membrane elements, 12,521 two-node cable elements and 1 one-node payload element. There are 29,200 nodes on the canopy. The fluid mechanics interface mesh has 2,140 nodes and 4,180 three-node triangular elements. The fluid mechanics mesh includes 178,270 nodes and 1,101,643 four-node tetrahedral elements. The dimensions of the computational domain, in ft, are $1{,}740 \times 1{,}740 \times 1{,}566$.

All computations are carried out using properties of air at standard sea-level conditions. In addition to moving the reference frame vertically with a reference descent speed, as originally proposed in [55], we move the mesh horizontally and vertically, with the average displacement rate for the structure. The horizontal motion of the mesh becomes particularly helpful when the parachute glides significantly. With a mesh that moves horizontally, we use the velocity form of the free-stream conditions also at the lateral boundaries.

All computations are carried out in a parallel computing environment, using PC clusters. The meshes are partitioned to enhance the parallel efficiency of the com-

putations, and mesh partitioning is based on the METIS [15] algorithm. In solving the linear equation systems involved at every nonlinear iteration, the GMRES search technique [25] is used with a diagonal preconditioner.

The stand-alone fluid mechanics computations are done in two parts. The first part uses the semi-discrete formulation given in [38]. We compute 1,000 time steps with a time-step size of 0.232 s and 7 nonlinear iterations per time step. The number of GMRES iterations per nonlinear iteration is 90. The second part uses the DSD/SST-TIP1 technique (see Remark 5), with the SUPG test function option WTSA (see Remark 1). The stabilization parameters used are those given by Eqs. (3.3)–(3.4), (3.5)–(3.8), and (3.9), with the τ_{SUGN2} term dropped from Eq. (3.3). The porosity model is HMGP-FG. We compute 600 time steps with a time-step size of 0.0232 s, 6 nonlinear iterations per time step, and 90 GMRES iterations per nonlinear iteration.

During the symmetric FSI and FSI computations, we use the SSTFSI-TIP1 technique (see Remarks 5 and 8), with the SUPG test function option WTSA. The stabilization parameters used are those given by Eqs. (3.3)–(3.4), (3.5)–(3.8), and (3.9), with the τ_{SUGN2} term dropped from Eq. (3.3). We use the SSP. The fully-discretized, coupled fluid and structural mechanics and mesh-moving equations are solved with the quasi-direct coupling technique (see Section 5.2 in [46]). We use selective scaling (see [46]), with the scale for the structure part set to 10. The time-step size is 0.0232 s. The number of nonlinear iterations per time step is 6. The number GMRES iterations per nonlinear iteration is 90 for the fluid+structure block, and 30 for the mesh-moving block.

5.3 Descent Speed Component Analysis

It was suggested in [32] that a methodology be developed for removing the dominant swinging component from the descent speed of the payload (U) to provide a better

comparison basis in parametric studies. Drawing conclusions about the effect of the OICL would be easier if the dominant swinging dynamic were not a factor in U. This contribution can vary due to damping throughout the computation and whether the trajectory of the swing is pendular or circular in nature. Here we will present and employ a method for removing the swinging component from U. Analyzing trajectory information from the computations, it was concluded that the canopy centroid is a very stable point for the canopy. We estimate the swinging component of the velocity, $(\boldsymbol{u}_s)_n$ as follows:

$$(\boldsymbol{u}_s)_n = \frac{1}{2\Delta t}\left(\frac{\boldsymbol{l}_{n+1}}{||\boldsymbol{l}_{n+1}||} - \frac{\boldsymbol{l}_{n-1}}{||\boldsymbol{l}_{n-1}||}\right)||\boldsymbol{l}_n||\,, \tag{5.1}$$

$$\boldsymbol{l}_n = (\boldsymbol{x}_p)_n - (\boldsymbol{x}_c)_n\,, \tag{5.2}$$

where $\boldsymbol{x}$ represents the position, the subscripts "p" and "c" stand for "payload" and "canopy centroid", and the subscript "n" is the time step. We subtract the vertical component of $(\boldsymbol{u}_s)_n$ from U and report $U - (\boldsymbol{u}_s)_z$. The significant horizontal speed makes a contribution to the overall drag of the parachute. In plots showing time-dependent behavior, it is difficult to compare strictly U values as a difference in glide ratio will have an effect on the apparent U. Therefore, we will also report the total velocity magnitude of the payload with the swinging component removed.

5.4 Results

Figure 5.1 shows that the skirt diameter restriction imposed by the OICL has removed the breathing oscillations, as expected.

We are interested in the effect of $W_\mathrm{p}/S_\mathrm{o}$ on the lateral movement of the parachute, primarily the horizontal speed developed in steady descent, which is an indication of stability. Decreased stability, which means more horizontal gliding, will lead to lower descent speeds that may seem to improve performance. When looking for the effect

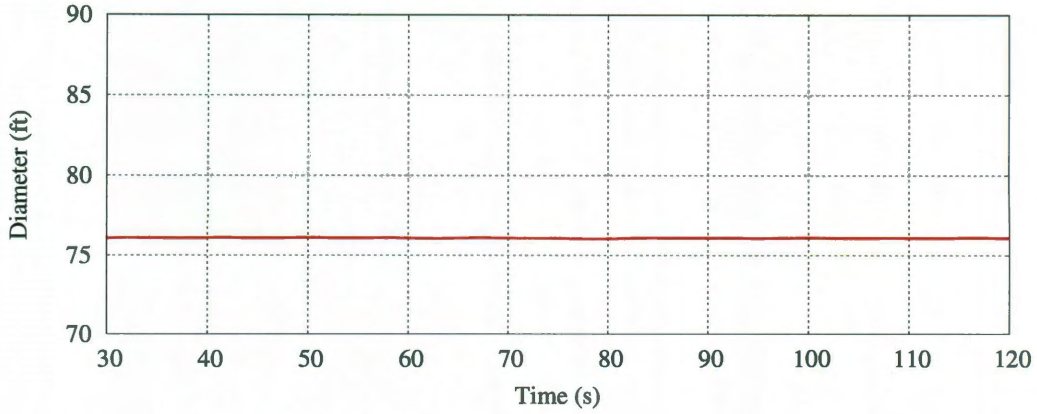


Figure 5.1: Skirt diameter for 76 ft OICL and $W_p/S_o = 0.650$.

of the OICL on drag production, total speed can sometimes provide a clearer picture than descent speed. Figures 5.2 and 5.3 show the payload descent speed and total speed with the swinging component removed.

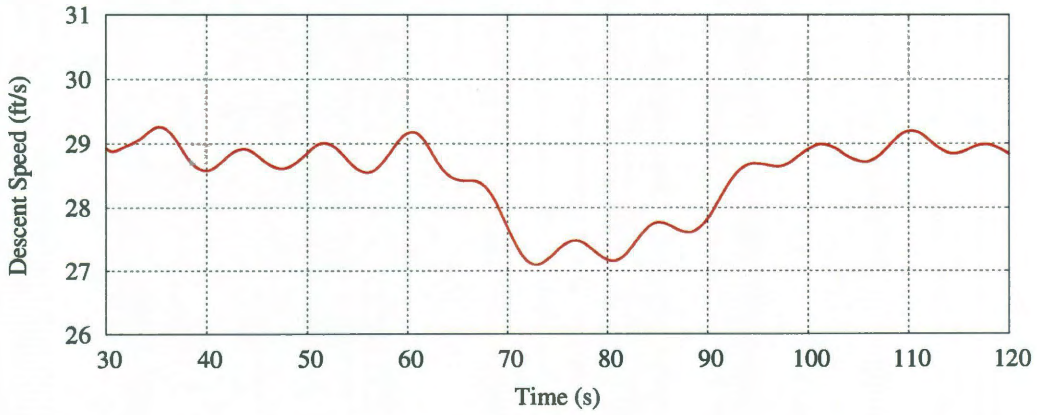


Figure 5.2: Payload descent speed (with the swinging component removed) for 76 ft OICL and $W_p/S_o = 0.650$.

In order to evaluate the lateral stability of the parachute, we non-dimensionalize the glide slope using the lift-to-drag ratio (L/D). Drag is defined as the component of canopy force in the direction of the relative wind. Lift is defined as the component of canopy force that is orthogonal to the relative wind vector in the plane formed by the relative wind vector and the parachute axis. Dividing the magnitude of the

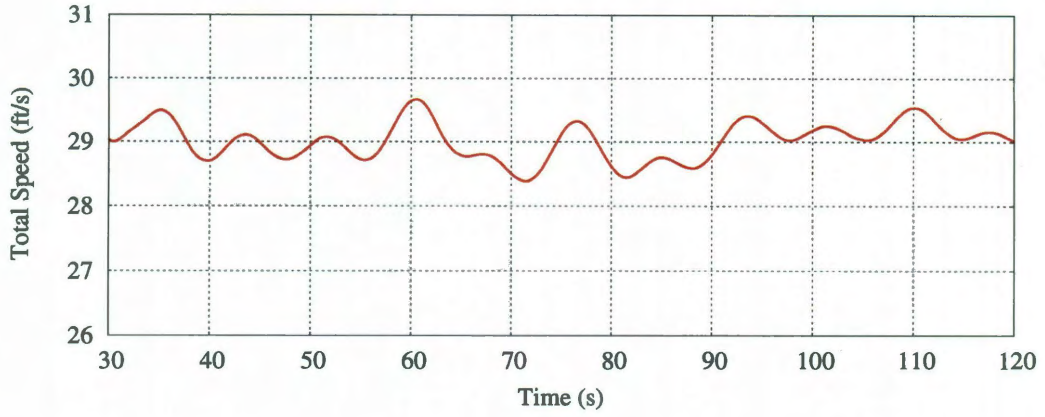


Figure 5.3: Payload total speed (with the swinging component removed) for 76 ft OICL and $W_p/S_o = 0.650$.

lift vector by the magnitude of the drag vector gives the L/D ratio. The L/D ratio provides a clearer indication of the parachute's tendency to glide independent of the descent speed. High L/D means low lateral stability and vice versa. Figure 5.4 shows the stability of the parachute in terms of L/D. Table 5.1 provides a summary of the

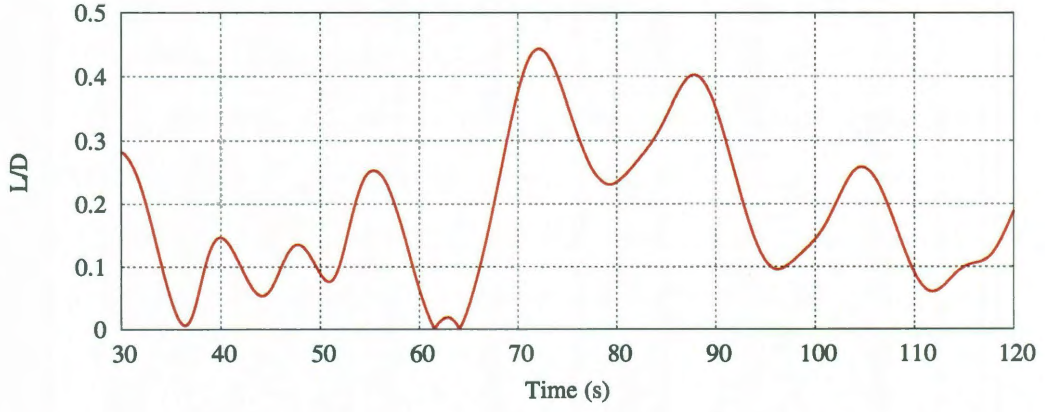


Figure 5.4: Lift-to-drag ratio (L/D) for 76 ft OICL and $W_p/S_o = 0.650$.

computational results.

While this analysis is relatively limited, it shows the general approach for parachute FSI modeling, data post-processing, and several common measures of parachute performance.

W_p/S_o	OICL	D_S (ft)	$U-(u_s)_z$	L/D
0.650	76	76.1	28.6	0.18

Table 5.1: Average values of the skirt diameter (D_S), $U-(u_s)_z$, and L/D for 76 ft OICL and $W_p/S_o = 0.650$.

Chapter 6

Parachute Cluster Computations

Parachute clusters, like the ones presented in this chapter, have several advantages over single parachutes when used to decelerate heavy payloads [16]. Several small parachutes are easier to manufacture and rig than a single large parachute with equivalent drag area. The filling time is also shorter for smaller parachutes. Employing a cluster of parachutes allows one parachute design to be utilized for a wide range of payload weights and offers redundancy if one parachute fails. Finally, clusters are generally more stable than single parachutes. Parachute clusters also have several disadvantages including drag reduction due to mutual interference and unequal loading due to asynchronous parachute opening. This chapter, after describing a technique for modeling the contact between parachutes in a cluster, presents an investigation of parachute cluster dynamics. Through this investigation, we seek to determine how the parameters representing the payload models and starting-conditions affect long-term cluster behavior.

6.1 Surface-Edge-Node Contact Tracking

There are a number of FSI computational challenges specific to parachute clusters, and contact between parachutes is one of the major challenges. For parachute FSI

computations, a contact algorithm is necessary to prevent the structural surfaces from coming closer than a predetermined minimum distance we would like to maintain to protect the quality of the fluid mechanics mesh between the structural surfaces. The Surface-Edge-Node Contact Tracking (SENCT) technique was introduced in [46] for this purpose. Two versions of the SENCT technique were proposed in [46]. In the SENCT-Force (SENCT-F) technique, the contacted node is subjected to penalty forces that are inversely proportional to the projection distances to the contacting surfaces, edges and nodes. In the SENCT-Displacement (SENCT-D) technique, the displacement of the contacted node is adjusted to correlate with the motion of the contacting surfaces, edges and nodes. The SENCT technique was described in more detail in [26], which included a number of test computations.

For FSI problems with incompatible fluid and structure meshes at the interface, it was proposed in Remark 1 of [50] to formulate the contact model based on the fluid mechanics mesh at the interface. This version of the SENCT was denoted with the option key "-M1". Basing the contact model on the fluid mechanics mesh instead of the structural mechanics mesh becomes significant (and helpful) when smoothing and homogenization techniques are employed to shelter the fluid mechanics mesh from the consequences of the geometric complexities of the structural surfaces.

6.2 SENCT-FC Technique

The SENCT-FC technique, which was introduced in [33], has some features in common with the SENCT-F technique but is more robust. Also, compared to the SENCT-F technique, the forces are applied in a conservative fashion, and the letter "C" in "FC" stands for "conservative". Later in this section, we will comment more on this aspect of the differences between SENCT-FC and SENCT-F (see Remark 10). The new technique is used as SENCT-FC-M1 in the computations reported here. It can

be seen as having three parts: contact detection, force representation, and solving the contact force equations. We describe those three parts from [33].

6.2.1 Contact Detection and Node Sets

To detect contact, we calculate the distance between a point and the closest point to it. The closest point is searched on surfaces, and the number of surfaces can be limited for efficiency. For example, we may choose to exclude self-contact. Here, we use the nodes on the fluid surface for contact detection, however this technique is applicable for any other kind of points (e.g. integration points).

The technique starts by finding the closest point in the same way as in the earlier SENCT techniques described in [50, 26]; i.e., the closet point on a node, edge or surface element. The node, edge, or surface element containing the closest point will be referred to as a "segment". Here we define $\mathbf{d}_A = \mathbf{x}_A^C - \mathbf{x}_A$ as the distance vector for each node and the closest point, which is represented by $\mathbf{x}^C$. In the case of $\|\mathbf{d}_A\| < \epsilon_A$, this node is in contact. The predetermined minimum distance is defined as follows:

$$\epsilon_A \equiv \epsilon_A^{\mathrm{S}} + \epsilon_A^{\mathrm{C}}, \tag{6.1}$$

where $\epsilon_A^{\mathrm{S}} \geq 0$ and $\epsilon_A^{\mathrm{C}} \geq 0$ are the length parameters for node A and the closest point, respectively. Figure 6.1 shows an example. The set of contacted nodes is defined as follows:

$$\eta_{\mathrm{D}} = \{A \mid \| \mathbf{d}_A\| < \epsilon_A, \ \forall A \in \eta\}, \tag{6.2}$$

where η represents all possible contacted nodes on the fluid surface. Each contacted node A has some contacting nodes, which are denoted by γ_A. All nodes in contact are represented as follows:

$$\eta_{\mathrm{C}} = \bigcup_{A \in \eta_{\mathrm{D}}} \{A + \gamma_A\}. \tag{6.3}$$

In general $\eta_D \subseteq \eta_C$. As a reverse relationship, we define ξ_A as the nodes contacted by node A. These set definitions are shown in Figures 6.1 and 6.2.

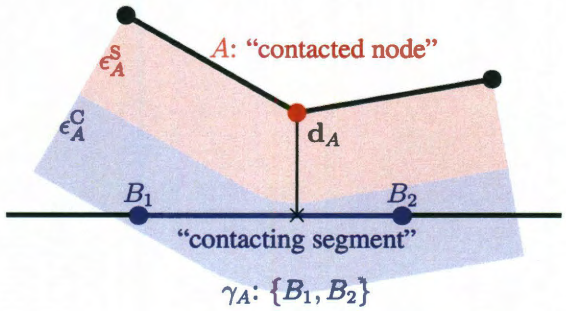


Figure 6.1: Contact detection and definitions of "contacted node" and "contacting-segment nodes".

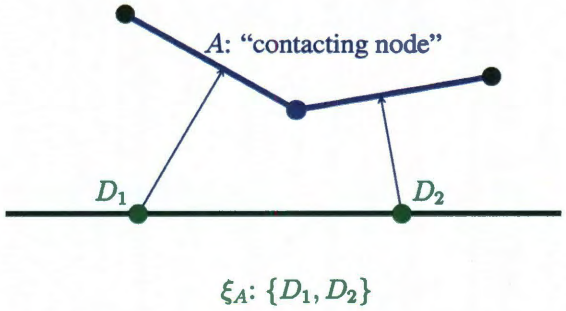


Figure 6.2: The set ξ_A contains the nodes contacted by node A.

6.2.2 Contact Force and Reaction Force

We introduce a virtual contact force $\boldsymbol{\varphi}_A$ for each contacted node A and a reaction force $\boldsymbol{\varphi}_B^R$ for node $B \in \gamma_A$. First, we model the force and the reaction forces as follows:

$$\boldsymbol{\varphi}_A = -\varphi_A \mathbf{n}_A, \tag{6.4}$$

$$\mathbf{n}_A = \frac{\mathbf{d}_A}{\|\mathbf{d}_A\|}, \tag{6.5}$$

and

$$\boldsymbol{\varphi}_B^{\mathrm{R}} = E_{BA}\varphi_A \mathbf{n}_A, \tag{6.6}$$

where E_{BA} is a scalar for each node $B \in \gamma_A$. Figure 6.3 shows the definition of the force and the reaction forces. The scalar values can be solved by using the following equations:

$$\boldsymbol{\varphi}_A + \sum_{B \in \gamma_A} \boldsymbol{\varphi}_B^{\mathrm{R}} = \mathbf{0}, \tag{6.7}$$

$$\sum_{B \in \gamma_A} \left(\mathbf{x}_B - \mathbf{x}_A^C\right) \times \boldsymbol{\varphi}_B^{\mathrm{R}} = \mathbf{0}. \tag{6.8}$$

The first equation is the balance of forces and the second equation is the no-moment condition. There is a unique set of solutions E_{BA} for Eqs. (6.7) and (6.8) in the cases of node, edge and triangle segments.

Remark 9 *We note that in the case of a node, edge and triangle segment, the scalar factor E_{BA} is the same as the shape function value $N_B\left(\mathbf{x}_A^C\right)$.*

The total force for each contacted node A is

$$\mathbf{f}_A = -\varphi_A \mathbf{n}_A + \sum_{D \in \xi_A} E_{AD}\varphi_D \mathbf{n}_D \quad \forall A \in \eta_{\mathrm{D}}, \tag{6.9}$$

and for the other nodes it is

$$\mathbf{f}_A = \sum_{D \in \xi_A} E_{AD}\varphi_D \mathbf{n}_D \quad \forall A \in \eta_{\mathrm{C}} - \eta_{\mathrm{D}}. \tag{6.10}$$

Figure 6.4 shows all the forces acting on node A.

Remark 10 *The contact forces are applied in SENCT-FC in a conservative fashion, because while we calculate the forces acting on a contacted node, we also calculate the corresponding reaction forces acting on the contacting nodes, as represented by*

Eqs. (6.4) and (6.6) and eventually by Eqs. (6.9) and (6.10). In SENCT-F on the other hand, only the forces acting on the contacted nodes are calculated, assuming that all the contacting nodes would also become contacted nodes during the search process, and therefore the contact forces are not applied in a conservative fashion.

We rewrite the i^{th} component of total force for node $A \in \eta_C$ as follows:

$$f_{Ai} = \left(E_{AC}\delta_{ij} - \delta_{AC}\delta_{ij}\right)\delta_{CD}n_{Dj}\varphi_D, \tag{6.11}$$

where we use the summation convention ($C, D \in \eta_D$ and $j = 1, \cdots, n_{sd}$). With the matrix-vector notation, the above equation becomes

$$\mathbf{F} = \mathbf{QV\Phi}, \tag{6.12}$$

where

$$\mathbf{F} = [f_{Ai}], \tag{6.13}$$

$$\mathbf{Q} = [E_{AC}\delta_{ij} - \delta_{AC}\delta_{ij}], \tag{6.14}$$

$$\mathbf{V} = [\delta_{CD}n_{Dj}], \tag{6.15}$$

$$\mathbf{\Phi} = [\varphi_D]. \tag{6.16}$$

6.2.3 Solving for the Contact Force

We use the following equation for the contacted node A:

$$\mathbf{d}_A \cdot \mathbf{d}_A = \epsilon_A^2. \tag{6.17}$$

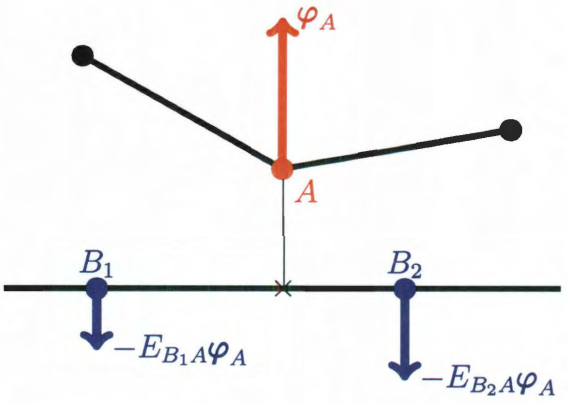


Figure 6.3: Contact force for the contacted node A and the reaction forces for the contacting nodes γ_A.

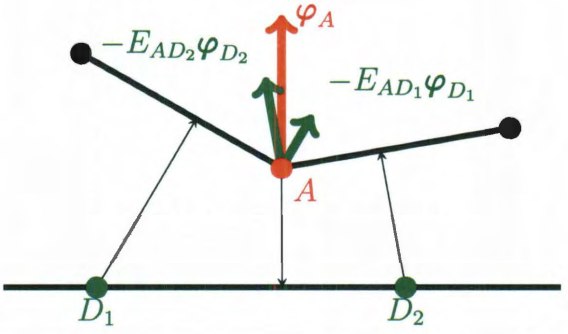


Figure 6.4: All the forces acting on node A. Node A is both "contacted" and "contacting".

For implementation convenience, we use block-iterative coupling between Eq. (6.17) and the fluid+structure block of the FSI system. In that framework, the block corresponding to Eq. (6.17) becomes:

$$\mathbf{n}_A^i \cdot \left(\left(\frac{\partial \mathbf{d}_A}{\partial \mathbf{x}_1} \right)^i \frac{\partial \mathbf{x}_1}{\partial \mathbf{x}_2} \left(\frac{\partial \mathbf{x}_2}{\partial \mathbf{F}_2} \right)^i \frac{\partial \mathbf{F}_2}{\partial \mathbf{F}_1} (\Delta \mathbf{F})^i \right) = \frac{\epsilon_A^2 - \|\mathbf{d}_A^i\|^2}{2 \|\mathbf{d}_A^i\|}, \tag{6.18}$$

where subscripts "1" and "2" represent the fluid and structure, respectively, and superscript i denotes the i^{th} nonlinear iteration. In the case of compatible fluid and

structure meshes at the interface, $\frac{\partial \mathbf{x}_1}{\partial \mathbf{x}_2}$ and $\frac{\partial \mathbf{F}_2}{\partial \mathbf{F}_1}$ are identity matrices. After solving for this contact force, we form the total force and apply it to the structure as an external force, then solve the fluid+structure block.

We describe each term in more detail below. The closest point can be expressed as follows:

$$\mathbf{x}_A^C = \sum_{B \in \gamma_A} N_B \left(\mathbf{x}_A^C\right) \mathbf{x}_B. \tag{6.19}$$

We define $H_{AB} \equiv N_B \left(\mathbf{x}_A^C\right)$, and the i^{th} component of the distance vector for $A \in \eta_{\mathrm{D}}$ can be written as

$$d_{Ai} = \left(H_{AB}\delta_{ij} - \delta_{AB}\delta_{ij}\right) x_{Bj}, \tag{6.20}$$

where $B \in \eta_{\mathrm{C}}$. With the matrix-vector notation, it becomes

$$\mathbf{D} = \mathbf{S}\mathbf{X}, \tag{6.21}$$

where

$$\mathbf{D} = \left[d_{Ai}\right], \tag{6.22}$$

$$\mathbf{S} = \left[\left(H_{AB}\delta_{ij} - \delta_{AB}\delta_{ij}\right)\right], \tag{6.23}$$

$$\mathbf{X} = \left[x_{Bj}\right]. \tag{6.24}$$

Thus,

$$\left(\frac{\partial \mathbf{D}}{\partial \mathbf{X}}\right) = \mathbf{S}. \tag{6.25}$$

Because of Remark 9, $\mathbf{S}$ is the transpose of $\mathbf{Q}$. Here we define three more matrices:

$$\mathbf{C} = \frac{\partial \mathbf{x}_1}{\partial \mathbf{x}_2}, \tag{6.26}$$

$$\mathbf{Z} = \frac{\partial \mathbf{x}_2}{\partial \mathbf{F}_2}, \tag{6.27}$$

$$\mathbf{B} = \frac{\partial \mathbf{F}_2}{\partial \mathbf{F}_1}. \tag{6.28}$$

Thus, we obtain the following equation system:

$$\left[\left(\mathbf{Q}^i\mathbf{V}^i\right)^T\left(\mathbf{C}\mathbf{Z}^i\mathbf{B}\right)\left(\mathbf{Q}^i\mathbf{V}^i\right)\right]\Delta\mathbf{\Phi}^i = \mathbf{\Psi}^i, \tag{6.29}$$

where

$$\mathbf{\Psi}^i = \left[\frac{\epsilon_A^2 - \|\mathbf{d}_A^i\|^2}{2\|\mathbf{d}_A^i\|}\right]. \tag{6.30}$$

We approximate $\mathbf{Z}^i$ by $\beta\Delta t^2\mathbf{M}^{-1}$, where $\mathbf{M}$ is the mass matrix and β is part of the Hilber–Hughes–Taylor [9] scheme. In the case of uncoupled multiple contacts, because of the nodal ordering we select, the coefficient matrix multiplying $\Delta\mathbf{\Phi}^i$ is in block-diagonal form. We split it into the block-diagonal matrices and solve each block directly by using LAPACK. Then, we apply the force $\mathbf{B}\mathbf{Q}^i\mathbf{V}^i\Delta\mathbf{\Phi}^i$ to the structure just like an external force.

Remark 11 *In the case of node-to-node contact, the forces of the two contacted nodes are not linearly independent. Therefore, we exclude one of the equations.*

6.3 Two-Parachute Cluster Computations

The objective in these two-parachute cluster computations is to determine how the parameters representing the payload models and starting-conditions affect long-term cluster dynamics. The parachute clusters reported in this paper are used with a 19,200 lb payload. The parachute geometry is described in detail in Chapter 1.

The parameters selected for testing are the payload-model configurations and initial coning angles (θ_{INIT}) and parachute diameters (D_{INIT}). For a definition of coning angle, see Figure 7.3. We also investigate two scenarios to approximate the conditions immediately after parachute disreefing. This is explained in more detail in a later paragraph. A summary of the computations is shown in Table 6.1. In all cases, the θ_{INIT} is the same for both parachutes.

The first set of computations investigates the effect of the payload model. In drop tests, the parachutes are connected to a rectangular pallet that is weighted to represent the mass and inertial properties of a proposed crew capsule. The preliminary parachute cluster computations reported in [34] modeled the payload as a point mass located at the confluence of the risers. We will refer to this as the payload at the confluence (PAC) configuration. Two new computational payload models were created to see how they would influence parachute behavior. The payload lower than the confluence (PLC) configuration adds another cable element below the confluence and models the payload as a point mass at the location of the pallet center of gravity. The payload as a truss element (PTE) configuration further enhances the model by distributing the payload mass at 9 different points to match the mass, center of gravity, and six components of the inertia tensor of the pallet. This is accomplished by adding 5 cable elements and 26 truss elements below the confluence (see [18]). Figure 6.5 shows the PTE configuration. All of the payload comparison computations use $\theta_{\text{INIT}} = 35°$.

The second set of computations investigates the effect of θ_{INIT}. Three values of

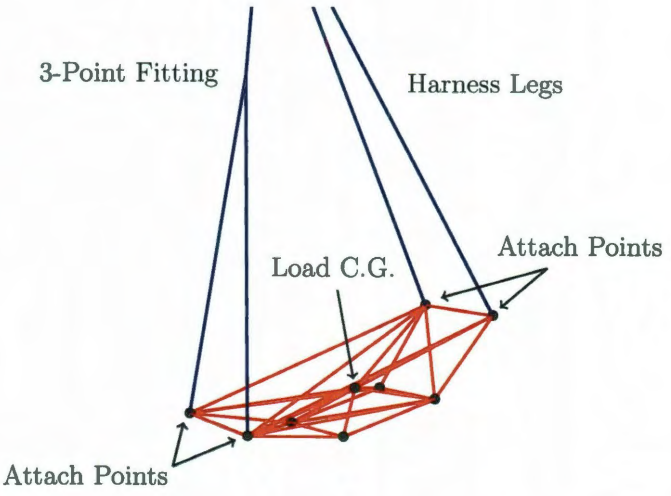


Figure 6.5: PTE configuration showing the cable (blue), truss (red), and payload (black) elements.

θ_{INIT} are tested: 15°, 25°, and 35°. It should be noted that 35° is greater than the θ values seen in drop tests. The average θ during normal descent is around 15°, and the maximum θ does not usually exceed 25°. We use $\theta_{\text{INIT}} = 35°$ only to cause a large perturbation in order to analyze the dynamic response of the parachute cluster. All of the θ_{INIT} comparison computations use the PTE configuration.

The parachute described in this paper uses a reefing technique to permit incremental opening of the canopy. The parachute skirt is initially constricted by reefing lines and the reefing lines are cut at preselected time intervals to allow the canopy to "disreef" to larger diameters. In the third set of computations, we compute two scenarios to analyze how conditions immediately after disreefing could have an effect on long-term dynamics. In the first scenario, which we call "simulated disreef", $\theta_{\text{INIT}} = 10°$, and for both parachutes $D_{\text{INIT}} = 70$ ft. These values represent the approximate θ during final disreefing and the average minimum D during nominal descent. The second scenario represents an "asynchronous disreef" by using for one parachute $D_{\text{INIT}} = 70$ ft, and for the other $D_{\text{INIT}} = 90$ ft. These values represent the average minimum and maximum parachute diameters during nominal descent, respectively. Both scenarios use the PTE configuration.

Payload Model	θ_{INIT} (°)	D_{INIT} (ft)	
		P_1	P_2
PAC	35	80	80
PLC	35	80	80
PTE	35	80	80
PTE	15	80	80
PTE	25	80	80
PTE	10	70	70
PTE	35	70	90

Table 6.1: Summary of all parachute cluster computations for different combinations of payload models, initial coning angles (θ_{INIT}), and parachute diameters (D_{INIT}). The values tabulated for θ_{INIT} apply to both parachutes, "P_1" and "P_2". The acronyms PAC, PLC and PTE represent payload at the confluence, payload lower than the confluence, and payload as a truss element.

6.3.1 Starting Conditions

We first build a starting condition for a single parachute. We begin with a parachute shape obtained with the symmetric FSI computation reported in [55]. We do another symmetric FSI computation with a horizontal inflow velocity of 24.0 $\sin(\theta_{\text{INIT}})$ ft/s. This results in an angle of attack of θ_{INIT}, and we compute for three breathing cycles. We use the parachute shape and position corresponding to the time when the parachute skirt diameter is at its average value and assemble the cluster structural mechanics mesh with the parachutes at θ_{INIT}. After that we generate a fluid mechanics mesh. With the cluster mesh, holding the parachute shapes and positions fixed, we first do a fluid mechanics computation. The inflow velocity is 31.0 ft/s. Next, we do a fluid mechanics computation with a prescribed, time-dependent shape for both parachutes. The time-dependent shape comes from the single-parachute symmetric FSI computation carried out earlier at an angle of attack of θ_{INIT}. We use the solution from the fluid mechanics computation with prescribed parachute motion as the

starting condition for the FSI computation.

6.3.2 Computational Conditions

The structural mechanics mesh for a two-parachute cluster has 61,443 nodes, 52,000 four-node quadrilateral membrane elements, and 25,042 two-node cable elements. There are 58,400 nodes on the canopies. The number of cable and payload elements can change depending on the payload model. The fluid mechanics interface mesh has 2,140 nodes and 4,180 three-node triangular elements. The fluid mechanics mesh is cylindrical with a diameter of 1,740 ft and a height of 1,566 ft. It consists of four-node tetrahedral elements, while the fluid interface mesh consists of three-node triangular elements. The number of nodes and elements are given in Table 6.2.

Structure		*nn*	61,443
	Membrane	*ne*	52,000
	Cable	*ne*	25,042
	Payload	*ne*	1
	Interface	*nn*	58,400
		ne	52,000
Fluid	Interface	*nn*	4,280
		ne	8,360
	Volume (15°, 80/80 ft)	*nn*	197,288
		ne	1,210,349
	Volume (25°, 80/80 ft)	*nn*	280,601
		ne	1,739,739
	Volume (35°, 80/80 ft)	*nn*	289,679
		ne	1,797,003
	Volume (10°, 70/70 ft)	*nn*	352,861
		ne	2,199,472
	Volume (35°, 70/90 ft)	*nn*	289,221
		ne	1,795,542

Table 6.2: Number of nodes and elements for the two-parachute clusters before any payload modifications. Here *nn* and *ne* are number of nodes and elements, respectively. The fluid mechanics volume mesh is tabulated for different combinations of θ_{INIT} and D_{INIT} values. The PLC configuration has 1 more structure node and 1 more cable element. The PTE configuration has 10 more structure nodes, 5 more cable elements, 26 more truss elements, and 8 more payload elements.

All computations are carried out using air properties at standard sea-level conditions. In addition to moving the reference frame vertically with a reference descent speed, as originally proposed in [55], we move the mesh horizontally and vertically, with the average displacement rate for the structure. The horizontal motion of the mesh becomes particularly helpful when the parachute glides significantly. With a mesh that moves horizontally, we use the velocity form of the free-stream conditions also at the lateral boundaries.

All computations are carried out in a parallel computing environment using PC clusters. The meshes are partitioned to enhance the parallel efficiency of the computations, and mesh partitioning is based on the METIS [15] algorithm. In solving the linear equation systems involved at every nonlinear iteration, the GMRES search technique [25] is used with a diagonal preconditioner.

The stand-alone fluid mechanics computations are done in two parts. The first part uses the semi-discrete formulation given in [38]. We compute 1,000 time steps with a time-step size of 0.232 s and 7 nonlinear iterations per time step. The number of GMRES iterations per nonlinear iteration is 90. The second part uses the DSD/SST-TIP1 technique (see Remark 5), with the SUPG test function option WTSA (see Remark 1). The stabilization parameters used are those given by Eqs. (3.3)–(3.4), (3.5)–(3.8), and (3.9), with the τ_{SUGN2} term dropped from Eq. (3.3). The porosity model is HMGP-FG. We compute 600 time steps with a time-step size of 0.0232 s, 6 nonlinear iterations per time step, and 90 GMRES iterations per nonlinear iteration.

For the fluid mechanics computations with prescribed, time-dependent shapes, we again use the DSD/SST-TIP1 technique, with the same SUPG test function option and stabilization parameters as those described above. We compute roughly 300 time steps with a time-step size of 0.0232 s, 6 nonlinear iterations per time step, and 90 GMRES iterations per nonlinear iteration.

For FSI computations, we use the SSTFSI-TIP1 technique (see Remarks 5 and

8), again with the same SUPG test function option and stabilization parameters as those described above. The fully-discretized, coupled fluid and structural mechanics and mesh-moving equations are solved with the quasi-direct coupling technique (see Section 5.2 in [46]). The time-step size is 0.0232 s, and the number of nonlinear iterations per time step is 6. The porosity model is HMGP-FG. We use SSP. We use selective scaling (see [46]), with the scale for the structure part set to 100. The SENCT-FC contact algorithm is used with $\epsilon_A^{\mathrm{S}} = \epsilon_A^{\mathrm{C}} = 1.45$ ft, which is approximately equal to the radial distance between the valley nodes and the outermost part of the sails at the parachute skirt. The number of GMRES iterations per nonlinear iteration is for most of the time steps 140 for the fluid+structure block, and 30 for the mesh-moving block. When the parachutes are close to each other, the number of GMRES iterations per nonlinear iteration for the fluid+structure block is increased as needed to control the residuals, especially those corresponding to the structural mechanics part. The maximum number of GMRES iterations used per nonlinear iteration for the fluid+structure block is 1,400.

We compute each parachute cluster for a total of about 75 s, and we remesh as needed to preserve mesh quality. The frequency of remeshing varies for each computation and usually depends on how often the parachutes collide, how much the cluster rotates about the vertical axis, and how much each parachute rotates about its own axis. Depending on the computation, remeshing is needed every 170 to 370 time steps.

6.3.3 Results

The critical measure of performance for the parachute system described in this paper is the descent speed of the payload. The maximum payload descent speed ultimately determines if the system meets mission requirements. Another common measure of performance is the drag coefficient, which is calculated as $C_{\mathrm{D}} = W_{\mathrm{p}}/(S_{\mathrm{o}}q)$ where q is

dynamic pressure. Dynamic pressure is computed with the expression $q = \frac{1}{2}\rho U^2$.

Figures 6.6–6.9 show the computational results for the parachute cluster computations.

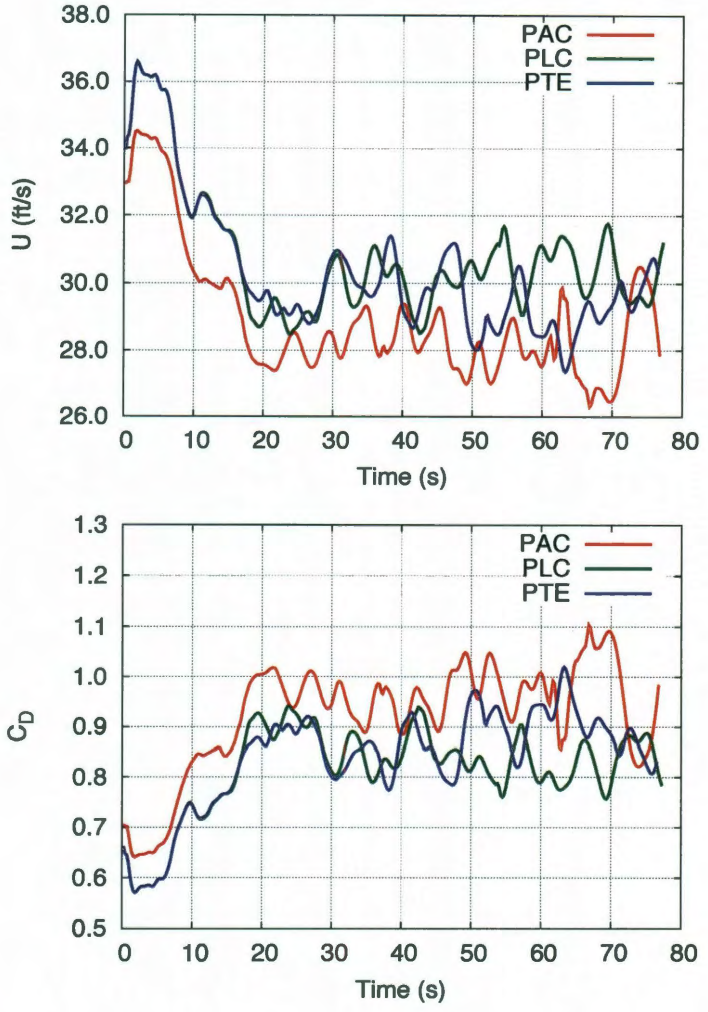


Figure 6.6: Cluster computations for different payload models and $\theta_{\text{INIT}} = 35°$.

The geometry of parachute clusters usually forces individual parachutes to fly at angles of attack that are higher than the angle of attack at which they would fly as single parachutes. If the forced angle of attack in the cluster is not a stable one for the parachutes, they tend to collide with each other as they attempt to reach an angle of attack that is stable. Figure 6.10 shows the contact between two parachutes from the

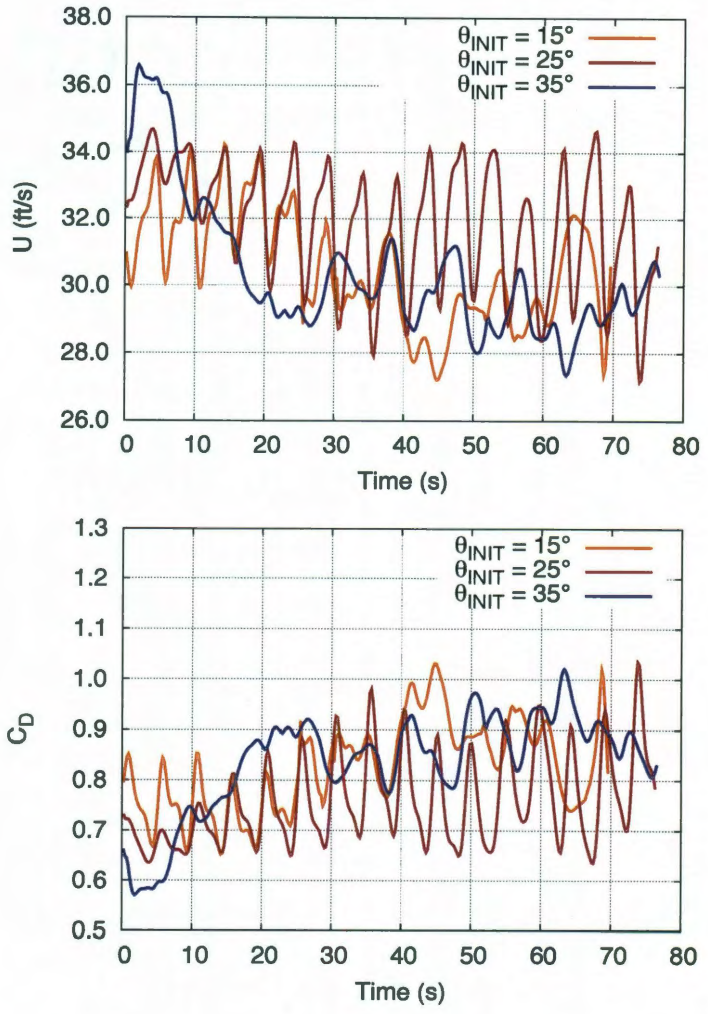


Figure 6.7: Cluster computations for PTE and different θ_{INIT} values.

asynchronous-disreef computation. Parachute clusters often experience reductions in drag due to this mutual interference between parachutes. The oscillatory motion of parachutes in the cluster and the frequency of collisions between parachutes can be used to characterize cluster stability. Figures 6.11–6.17 show the vent-separation distance ("L_{VS}") for all cluster computations. The horizontal black line on each plot shows the approximate vent-separation distance when the parachutes are in contact. Tables 6.3–6.5 summarize the payload descent speeds and drag coefficients for all of the cluster computations.

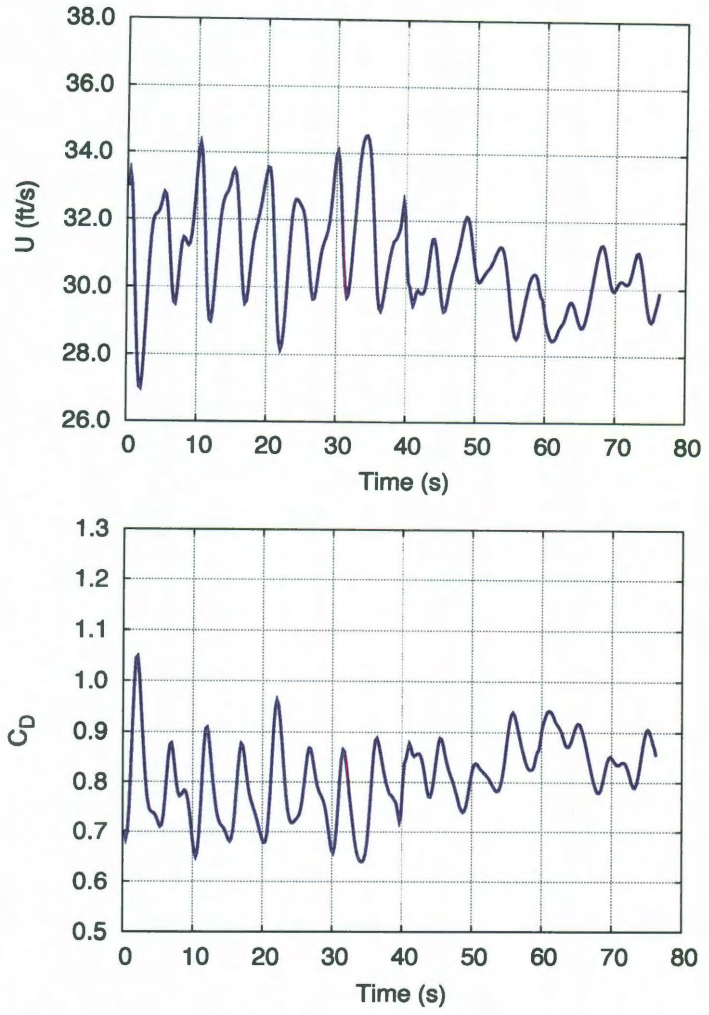


Figure 6.8: Cluster computations for simulated disreef.

One of the goals of this computational analysis is to assist parachute design engineers in determining which factors contribute to the payload descent speed oscillations seen in drop tests. For example, collisions between parachutes are usually associated with increased payload descent speed, but this is not always true. Previous analyses have also noted some correlation between parachute coning angles and payload descent speed. However, the correlation between these parameters is not strong enough to conclude that coning angle is the only, or even the most important, factor. The payload descent speed is composed of several overlapping frequencies caused by var-

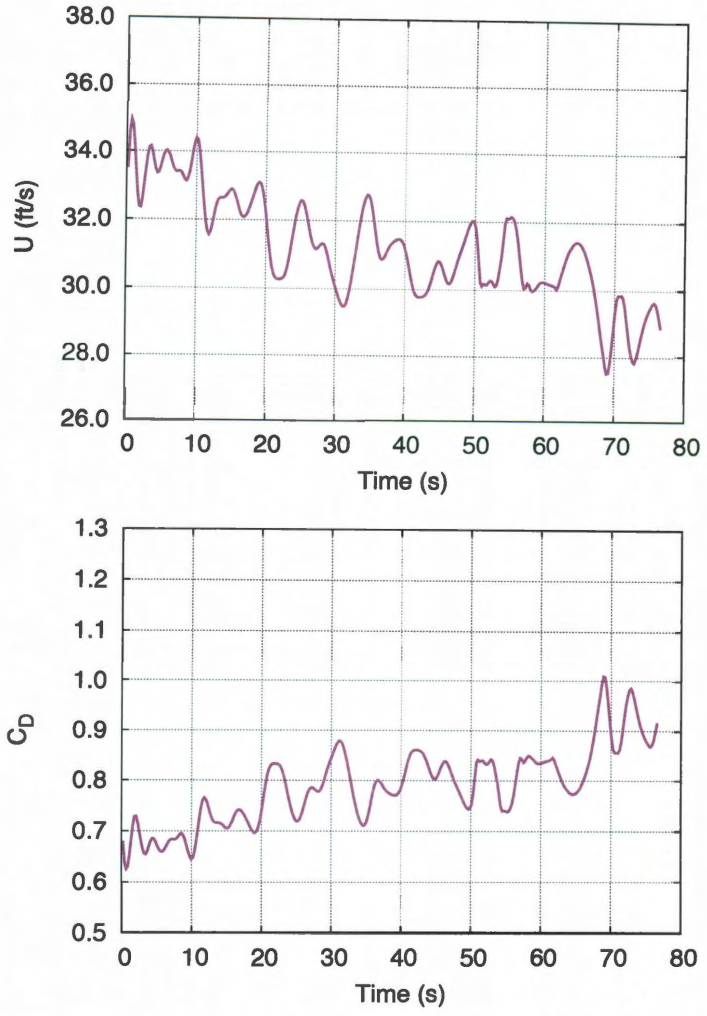


Figure 6.9: Cluster computations for asynchronous-disreef.

ious parachute dynamics. The overlap makes it very difficult to determine which individual parachute behaviors and parachute cluster behaviors are contributing to changes in payload descent speed. In order to address this complex problem, we have developed a technique to decompose the payload descent speed into components. This technique is described in Chapter 7.

Payload Model	U (ft/s)	C_D
PAC	28.1	0.97
PLC	30.1	0.85
PTE	29.5	0.88

Table 6.3: Average U and C_D for different payload models with $\theta_{INIT} = 35°$. Statistical analysis begins 20 s after the start of the computation.

θ_{INIT}	U (ft/s)	C_D
15°	29.9	0.86
25°	31.4	0.78
35°	29.5	0.88

Table 6.4: Average U and C_D for PTE and different values of θ_{INIT}. Statistical analysis begins 20 s after the start of the computation.

	U (ft/s)	C_D
Simulated Disreef	30.6	0.82
Asynchronous Disreef	30.8	0.81

Table 6.5: Average U and C_D for the disreef cases. Statistical analysis begins 5 s after the start of the computation for the simulated-disreef case, and 20 s after the start of the computation for the asynchronous-disreef case.

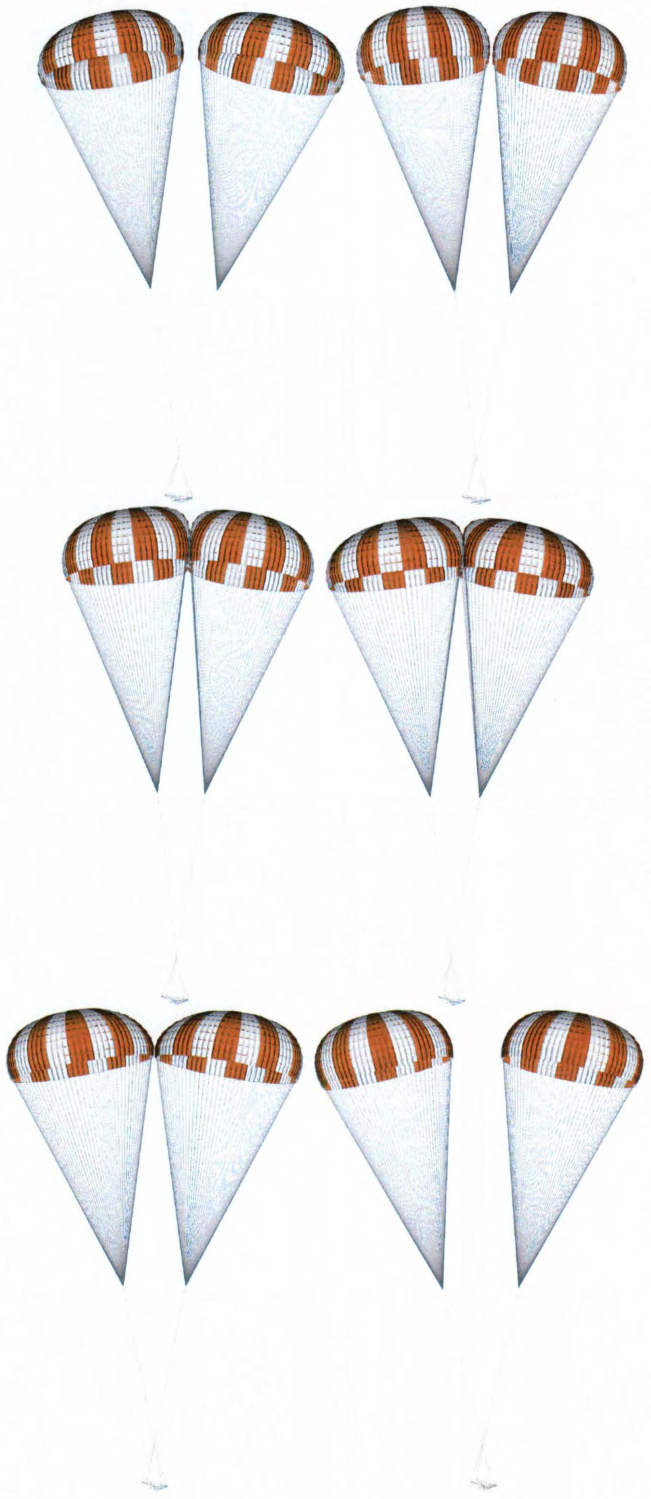


Figure 6.10: Parachutes from $t = 52.20$ s to $t = 58.00$ s, at 1.16 s intervals from left to right and top to bottom, during the asynchronous-disreef computation modeling the contact between parachutes.

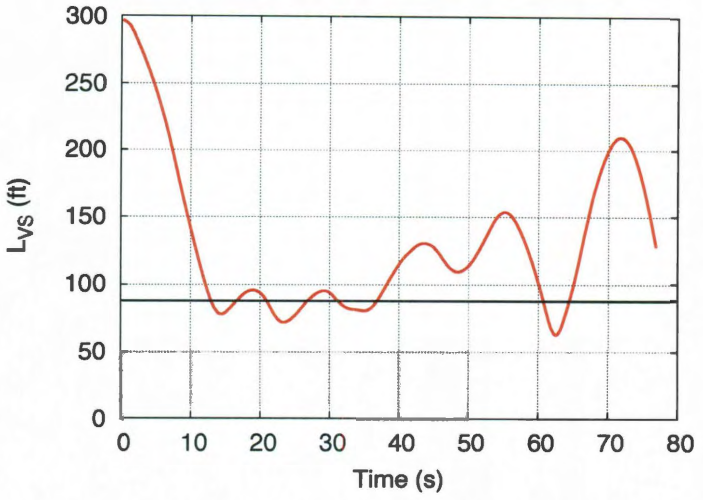


Figure 6.11: Vent-separation distance during the cluster computation with PAC and $\theta_{\text{INIT}} = 35°$.

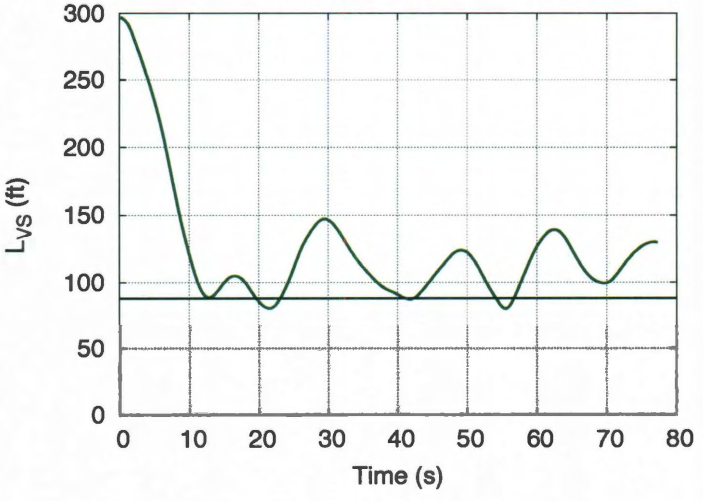


Figure 6.12: Vent-separation distance during the cluster computation with PLC and $\theta_{\text{INIT}} = 35°$.

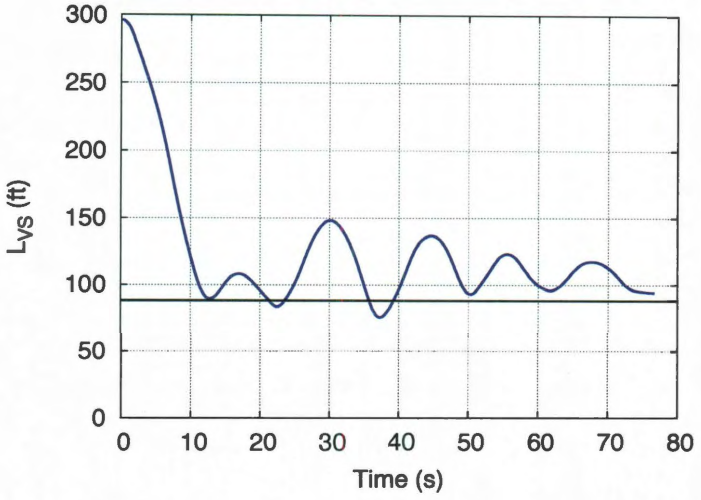


Figure 6.13: Vent-separation distance during the cluster computation with PTE and $\theta_{\text{INIT}} = 35°$.

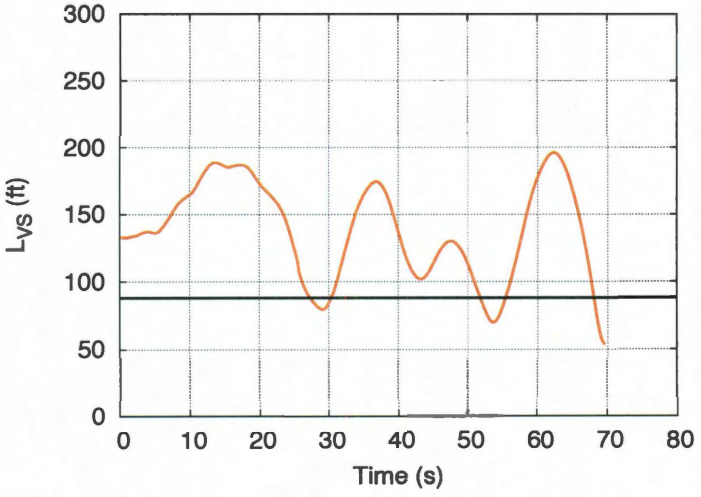


Figure 6.14: Vent-separation distance during the cluster computation with PTE and $\theta_{\text{INIT}} = 15°$.

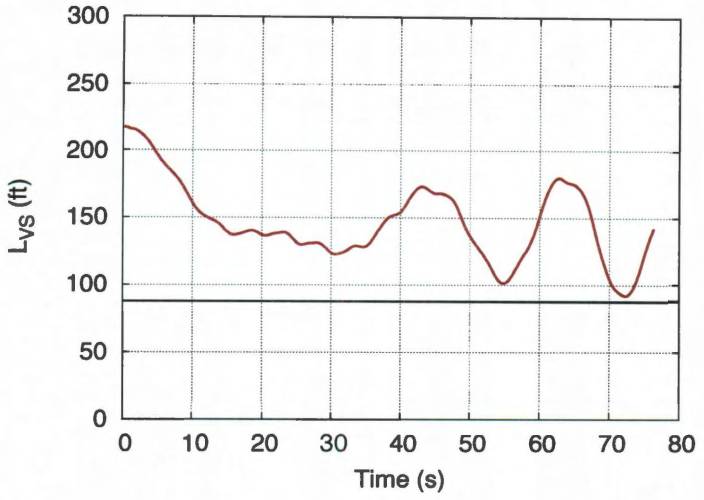


Figure 6.15: Vent-separation distance during the cluster computation with PTE and $\theta_{\text{INIT}} = 25°$.

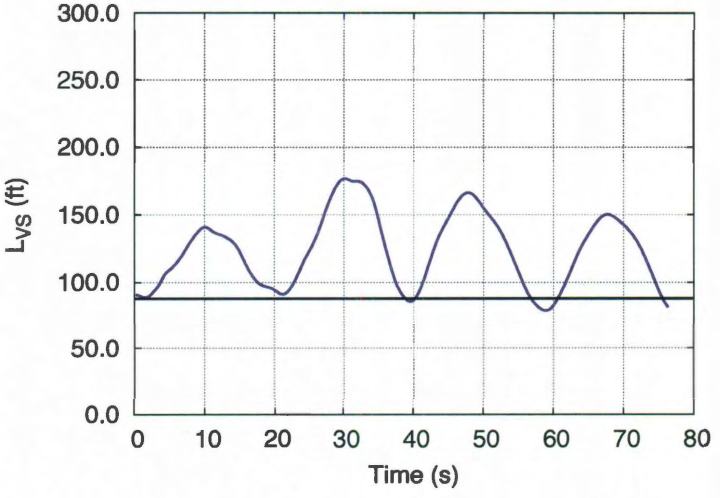


Figure 6.16: Vent-separation distance during the simulated-disreef cluster computation.

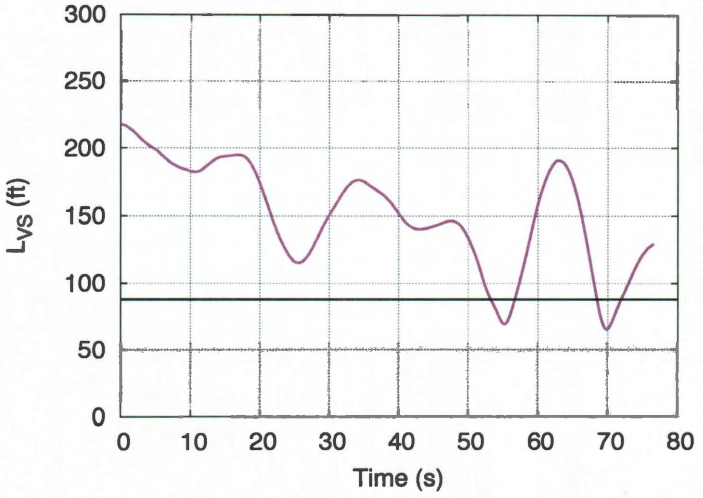


Figure 6.17: Vent-separation distance during the asynchronous-disreef cluster computation.

Chapter 7

Dynamical Analysis and Model Parameter Extraction

Dynamical analysis of the data coming from parachute FSI computations requires an approach that helps us make sense out of a large volume of time-dependent information generated by the computations. We need to extract and present the significant information in a way that would make it easier for parachute design engineers to make use of it. In addition to providing time histories of the aerodynamically significant quantities, such as the descent speed and forces, we may sometimes find it useful to show the various contributors to these quantities separately. We describe a special decomposition technique developed in [33] for parachute descent speed. We also describe a special technique developed in [33] for extracting from a parachute FSI computation model parameters that can be used in fast, approximate engineering analysis models for parachute dynamics. The specific parameters we are extracting are the added mass and the coefficient for the velocity-proportional aerodynamic force.

7.1 Parachute Descent Speed Decomposition

To better understand the descent speed fluctuations, we decompose the payload velocity into components based on geometric contributing factors. We carry out the analysis first for a single parachute, and then for a cluster of parachutes.

7.1.1 Single parachute

We represent the parachute in a spherical polar coordinate system. In that system, the payload is the origin, and the basis vectors are given in terms of the Cartesian basis vectors $\mathbf{e}_x$, $\mathbf{e}_y$ and $\mathbf{e}_z$ as follows:

$$\mathbf{g}_r = \sin\theta\cos\phi\ \mathbf{e}_x + \sin\theta\sin\phi\ \mathbf{e}_y + \cos\theta\ \mathbf{e}_z, \tag{7.1}$$

$$\mathbf{g}_\theta = \cos\theta\cos\phi\ \mathbf{e}_x + \cos\theta\sin\phi\ \mathbf{e}_y - \sin\theta\ \mathbf{e}_z, \tag{7.2}$$

$$\mathbf{g}_\phi = -\sin\phi\ \mathbf{e}_x + \cos\phi\ \mathbf{e}_y. \tag{7.3}$$

Here $\mathbf{g}_r$ and $\mathbf{g}_\theta$ represent the direction of the parachute axis and swinging, respectively, and $\mathbf{r} = r\mathbf{g}_r$ as shown in Figure 7.1. We separate the payload velocity $\mathbf{u}_\mathrm{p} \equiv \frac{\mathrm{d}\mathbf{x}_\mathrm{p}}{\mathrm{d}t}$ into its geometric ($\mathbf{u}_\mathrm{G}$) and aerodynamic ($\mathbf{u}_\mathrm{A}$) contributors as follows:

$$\mathbf{u}_\mathrm{p} = \mathbf{u}_\mathrm{G} + \mathbf{u}_\mathrm{A}, \tag{7.4}$$

$$\mathbf{u}_\mathrm{G} = \frac{\mathrm{d}\left(\mathbf{x}_\mathrm{p} - \mathbf{x}_\mathrm{A}\right)}{\mathrm{d}t}, \tag{7.5}$$

where $\mathbf{x}_\mathrm{A}$ is a reference point, meant to be selected to give us a good way of differentiating between the factors contributing to the payload velocity. Here we use the canopy centroid as that reference point. We define the relative position vector $\mathbf{r} = (\mathbf{x}_\mathrm{A} - \mathbf{x}_\mathrm{p})$, and obtain

$$\mathbf{u}_\mathrm{G} = -\dot{\mathbf{r}}. \tag{7.6}$$

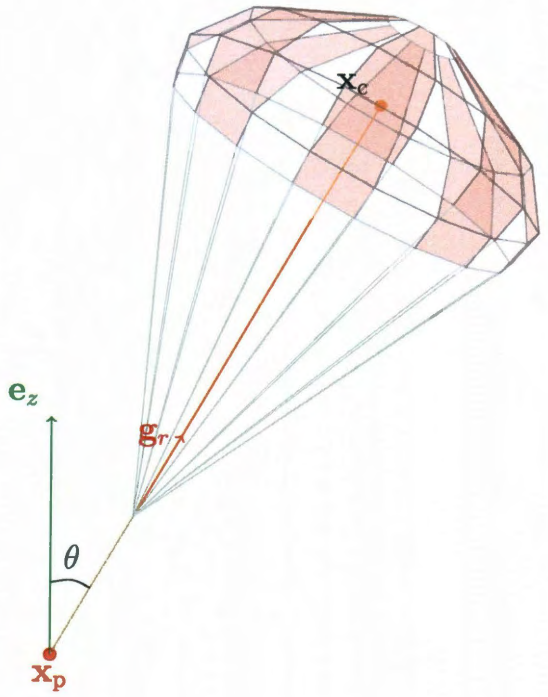


Figure 7.1: Parachute axis $\mathbf{g}_r$. Swinging angle θ.

The geometric contribution can be rewritten as follows:

$$\mathbf{u}_G = -\frac{\mathrm{d}(r\mathbf{g}_r)}{\mathrm{d}t} = -\dot{r}\mathbf{g}_r - r\frac{\mathrm{d}\mathbf{g}_r}{\mathrm{d}t} \tag{7.7}$$

$$= \underbrace{-\dot{r}\mathbf{g}_r}_{\mathbf{u}_B} \underbrace{-r\mathbf{g}_\theta\dot{\theta}}_{\mathbf{u}_S} \underbrace{-r\sin\theta\mathbf{g}_\phi\dot{\phi}}_{\mathbf{u}_C}, \tag{7.8}$$

where $\mathbf{u}_B$, $\mathbf{u}_S$ and $\mathbf{u}_C$ represent the parachute breathing, swinging and coning, respectively. We note that because the axes are orthogonal to each other, the decomposition becomes

$$\mathbf{u}_B = (\mathbf{u}_G \cdot \mathbf{g}_r)\mathbf{g}_r, \tag{7.9}$$

$$\mathbf{u}_S = (\mathbf{u}_G \cdot \mathbf{g}_\theta)\mathbf{g}_\theta, \tag{7.10}$$

$$\mathbf{u}_C = (\mathbf{u}_G \cdot \mathbf{g}_\phi)\mathbf{g}_\phi. \tag{7.11}$$

7.1.2 Cluster of Parachutes

We define $\overline{\mathbf{x}}_{\mathrm{A}}$ as

$$\overline{\mathbf{x}}_{\mathrm{A}} = \frac{1}{n_{\mathrm{para}}} \sum_{k=1}^{n_{\mathrm{para}}} (\mathbf{x}_{\mathrm{A}})_k, \tag{7.12}$$

where n_{para} is the number of parachutes. We again use a spherical polar coordinate systems with the payload being the origin and the basis vectors given as

$$\mathbf{g}_r = \sin\theta\cos\phi\ \mathbf{e}_x + \sin\theta\sin\phi\ \mathbf{e}_y + \cos\theta\ \mathbf{e}_z, \tag{7.13}$$

$$\mathbf{g}_\theta = \cos\theta\cos\phi\ \mathbf{e}_x + \cos\theta\sin\phi\ \mathbf{e}_y - \sin\theta\ \mathbf{e}_z, \tag{7.14}$$

$$\mathbf{g}_\phi = -\sin\phi\ \mathbf{e}_x + \cos\phi\ \mathbf{e}_y. \tag{7.15}$$

Here $\mathbf{g}_r$, $\mathbf{g}_\theta$ and $\mathbf{g}_\phi$ can be seen as the direction of the cluster axis, swinging and coning, and $\overline{\mathbf{r}} = \overline{r}\ \mathbf{g}_r$. Figure 7.2 shows the relationship between the Cartesian axes and the basis vectors. We also define individual spherical coordinate systems for the parachutes of the cluster. In each of those coordinate systems, the payload is the origin, and the basis vectors are given in terms of the basis vectors $\mathbf{g}_\theta$, $\mathbf{g}_\phi$ and $\mathbf{g}_r$:

$$(\mathbf{g}_r)_k = \sin\theta_k\cos\phi_k\mathbf{g}_\theta + \sin\theta_k\sin\phi_k\mathbf{g}_\phi + \cos\theta_k\mathbf{g}_r, \tag{7.16}$$

$$(\mathbf{g}_\theta)_k = \cos\theta_k\cos\phi_k\mathbf{g}_\theta + \cos\theta_k\sin\phi_k\mathbf{g}_\phi - \sin\theta_k\mathbf{g}_r, \tag{7.17}$$

$$(\mathbf{g}_\phi)_k = -\sin\phi_k\mathbf{g}_\theta + \cos\phi_k\mathbf{g}_\phi. \tag{7.18}$$

Here $(\mathbf{g}_r)_k$ is the axis direction for the k^{th} parachute, θ_k is the coning angle, and $(\mathbf{x}_{\mathrm{A}})_k - \mathbf{x}_{\mathrm{p}} \equiv \mathbf{r}_k = r_k(\mathbf{g}_r)_k$. Figure 7.3 shows the relationship between the cluster axes, the individual parachutes and $(\mathbf{g}_r)_k$.

From Eq. (7.12) and the definition of $\mathbf{r}_k$,

$$\mathbf{u}_{\mathrm{p}} = \dot{\overline{\mathbf{x}}}_{\mathrm{A}} - \dot{\overline{\mathbf{r}}}, \tag{7.19}$$

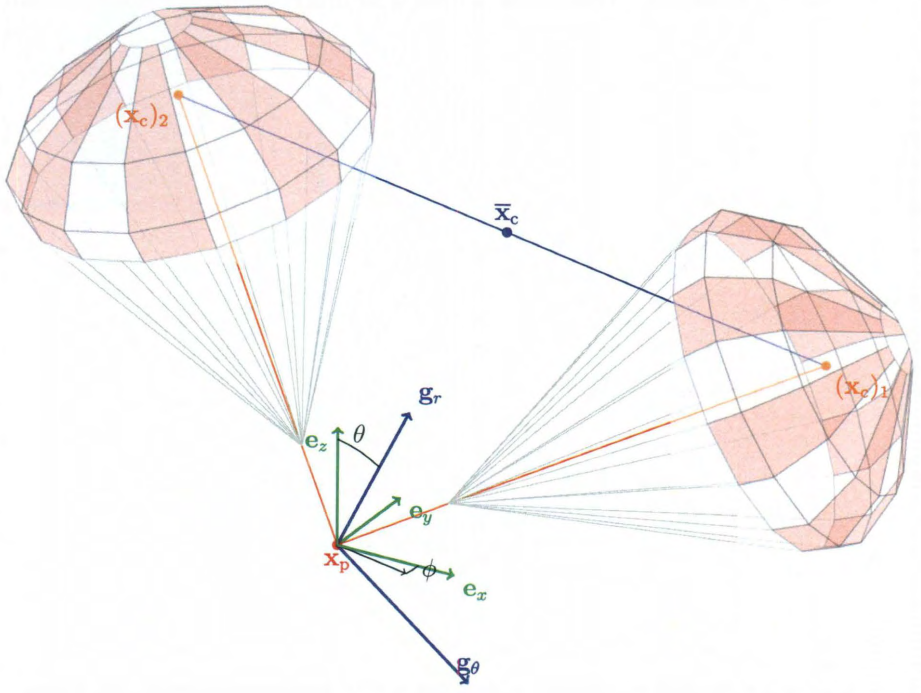


Figure 7.2: Direction of the cluster axis $\mathbf{g}_r$ and the cluster coning angle θ.

where

$$\overline{\mathbf{r}} = \frac{1}{n_{\text{para}}} \sum_{k=1}^{n_{\text{para}}} \mathbf{r}_k. \tag{7.20}$$

We define

$$\overline{\mathbf{u}}_{\text{G}} = -\dot{\overline{\mathbf{r}}}, \tag{7.21}$$

therefore

$$\mathbf{u}_{\text{p}} = \overline{\mathbf{u}}_{\text{A}} + \overline{\mathbf{u}}_{\text{G}}. \tag{7.22}$$

First we consider the cluster system itself. As we did for a single parachute, we can decompose the geometric contribution as follows:

$$\overline{\mathbf{u}}_{\text{G}} = \mathbf{u}_{\text{B}} + \mathbf{u}_{\text{S}} + \mathbf{u}_{\text{C}}, \tag{7.23}$$

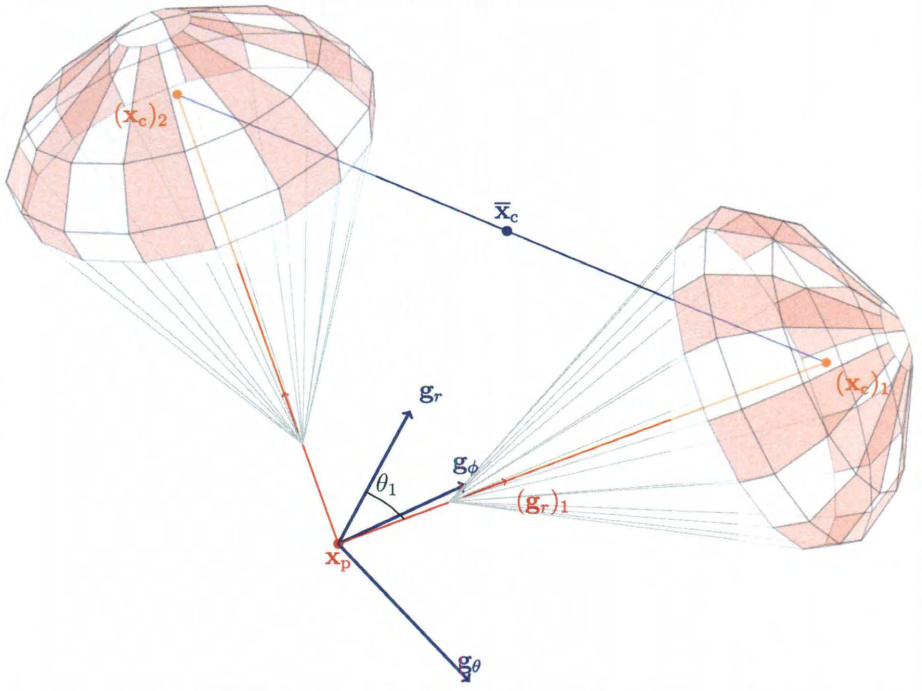


Figure 7.3: Direction of the parachute axis $(\mathbf{g}_r)_k$ and the individual parachute coning angle θ_k.

where $\mathbf{u}_B$, $\mathbf{u}_S$ and $\mathbf{u}_C$ represent the cluster system "breathing", swinging and coning:

$$\mathbf{u}_B = (\overline{\mathbf{u}}_G \cdot \mathbf{g}_r)\mathbf{g}_r, \tag{7.24}$$

$$\mathbf{u}_S = (\overline{\mathbf{u}}_G \cdot \mathbf{g}_\theta)\mathbf{g}_\theta, \tag{7.25}$$

$$\mathbf{u}_C = (\overline{\mathbf{u}}_G \cdot \mathbf{g}_\phi)\mathbf{g}_\phi. \tag{7.26}$$

Furthermore, we decompose the velocity into contributions from individual parachutes, which are written in terms of their breathing, "swinging" and coning parts:

$$(\mathbf{u}_G)_k = (\mathbf{u}_B)_k + (\mathbf{u}_S)_k + (\mathbf{u}_C)_k. \tag{7.27}$$

Here

$$(\mathbf{u}_B)_k = ((\mathbf{u}_G)_k \cdot (\mathbf{g}_r)_k)(\mathbf{g}_r)_k, \tag{7.28}$$

$$(\mathbf{u}_S)_k = ((\mathbf{u}_G)_k \cdot (\mathbf{g}_\theta)_k)(\mathbf{g}_\theta)_k, \tag{7.29}$$

$$(\mathbf{u}_C)_k = ((\mathbf{u}_G)_k \cdot (\mathbf{g}_\phi)_k)(\mathbf{g}_\phi)_k. \tag{7.30}$$

By definition,

$$\overline{\mathbf{u}}_G = \frac{1}{n_{\text{para}}} \sum_{k=1}^{n_{\text{para}}} (\mathbf{u}_G)_k. \tag{7.31}$$

We also define the averages of the parts given by Eqs. (7.28)–(7.30):

$$\overline{\mathbf{u}}_B = \frac{1}{n_{\text{para}}} \sum_{k=1}^{n_{\text{para}}} (\mathbf{u}_B)_k, \tag{7.32}$$

$$\overline{\mathbf{u}}_S = \frac{1}{n_{\text{para}}} \sum_{k=1}^{n_{\text{para}}} (\mathbf{u}_S)_k, \tag{7.33}$$

$$\overline{\mathbf{u}}_C = \frac{1}{n_{\text{para}}} \sum_{k=1}^{n_{\text{para}}} (\mathbf{u}_C)_k. \tag{7.34}$$

The cluster "breathing" comes from $((\overline{\mathbf{u}}_B)_k \cdot \mathbf{g}_r)\,\mathbf{g}_r$ and $((\overline{\mathbf{u}}_S)_k \cdot \mathbf{g}_r)\,\mathbf{g}_r$. We note from Eq. (7.18) that $\overline{\mathbf{u}}_C$ does not have a part in the direction of the cluster axis.

In summary,

$$\overline{\mathbf{u}}_G = \mathbf{u}_B + \mathbf{u}_S + \mathbf{u}_C = \overline{\mathbf{u}}_B + \overline{\mathbf{u}}_S + \overline{\mathbf{u}}_C. \tag{7.35}$$

We note that $\mathbf{u}_B$ and $\overline{\mathbf{u}}_B$, $\mathbf{u}_S$ and $\overline{\mathbf{u}}_S$, and $\mathbf{u}_C$ and $\overline{\mathbf{u}}_C$ represent things that are different from each other. In this paper we focus on the vertical components of all

these velocities, which, for example for the cluster, can be expressed as follows:

$$\mathbf{u}_B \cdot \mathbf{e}_z = (((\overline{\mathbf{u}}_B)_k \cdot \mathbf{g}_r) + ((\overline{\mathbf{u}}_S)_k \cdot \mathbf{g}_r))(\mathbf{g}_r \cdot \mathbf{e}_z), \tag{7.36}$$

$$\mathbf{u}_S \cdot \mathbf{e}_z = (((\overline{\mathbf{u}}_B)_k \cdot \mathbf{g}_\theta) + ((\overline{\mathbf{u}}_S)_k \cdot \mathbf{g}_\theta) + ((\overline{\mathbf{u}}_C)_k \cdot \mathbf{g}_\theta)) (\mathbf{g}_\theta \cdot \mathbf{e}_z), \tag{7.37}$$

$$\mathbf{u}_C \cdot \mathbf{e}_z = 0. \tag{7.38}$$

7.1.3 Results

Figures 7.4–7.10 show, for all cluster computations, the decomposition of the descent speed. For the cluster, the decomposition is in terms of $\overline{\mathbf{u}}_A$ and the "breathing" and swinging contributors of the descent speed. Top plots represent that. For the average of the individual parachute contributions, it is in terms of their breathing, "swinging" and coning parts. Bottom plots represent that.

From the top plots, we note that the contribution from $\mathbf{u}_B$ is always greater than the contribution from $\mathbf{u}_S$. Furthermore, cluster swinging does not appear to be a major contributor to payload descent speed oscillations. Most of the oscillations are coming from the motion of the individual parachutes. Looking at the bottom plots, we observe that for all of the computations with $\theta_{INIT} = 35°$, the largest contribution comes from $\overline{\mathbf{u}}_S$. For the computations with smaller θ_{INIT}, the $\overline{\mathbf{u}}_B$ contribution is dominant initially, and the $\overline{\mathbf{u}}_S$ contribution grows over time. The $\theta_{INIT} = 25°$ case seems to have fairly equal contributions from both $\overline{\mathbf{u}}_S$ and $\overline{\mathbf{u}}_B$. The contribution from the individual parachute coning, $\overline{\mathbf{u}}_C$, is the smallest.

We further analyze a particularly interesting result from the cluster computation with PTE and $\theta_{INIT} = 35°$ to better understand the parachute behavior that is causing the anomaly. Approximately 38 s into the computation, there is a large positive contribution from $(\mathbf{u}_S)_2$ followed by a large negative contribution (see Figure 7.11). In comparison, $(\mathbf{u}_S)_1$ remains fairly constant over the same time period. From the

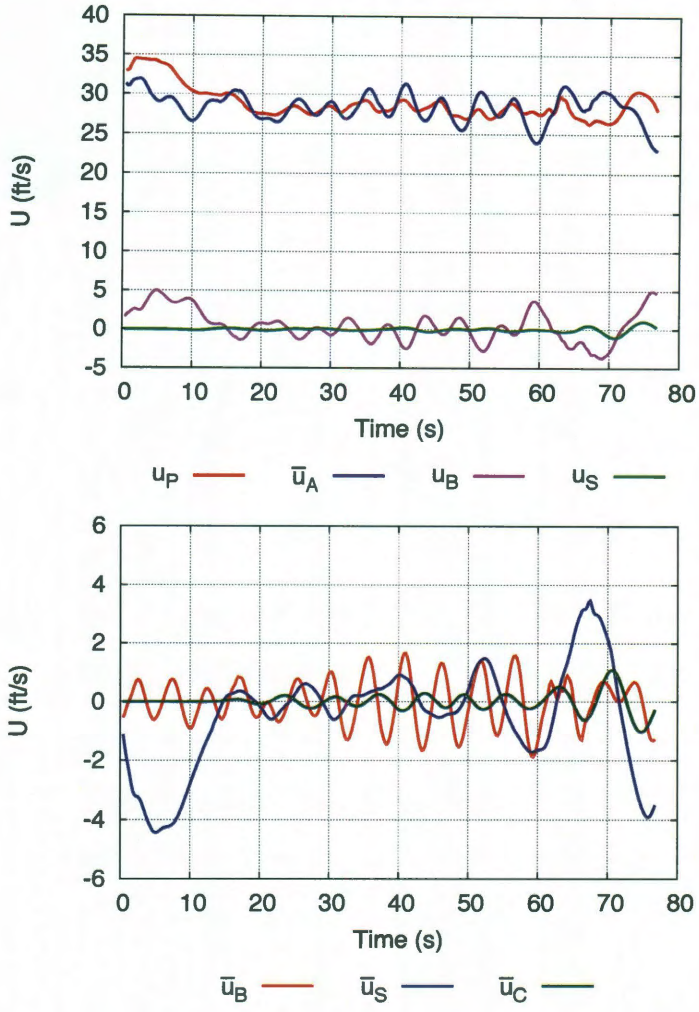


Figure 7.4: Decomposition of the descent speed for the cluster computation with PAC and $\theta_{\text{INIT}} = 35°$.

plots of payload and canopy-centroid descent speeds shown in Figure 7.12, we see that Parachute 2 initially has a much higher descent speed than Parachute 1. After about the 45 s mark, the descent speed for Parachute 2 becomes much lower than Parachute 1. The payload descent speed is relatively shielded because the fluctuations for Parachutes 1 and 2 are opposite each other and the average of the canopy speed fluctuations is small. We have determined that the large changes in $(\mathbf{u}_S)_2$ occur as Parachute 2 flies around Parachute 1 in a coning motion. Meanwhile, Parachute 1 descends in a relatively straight path and presents a larger projected

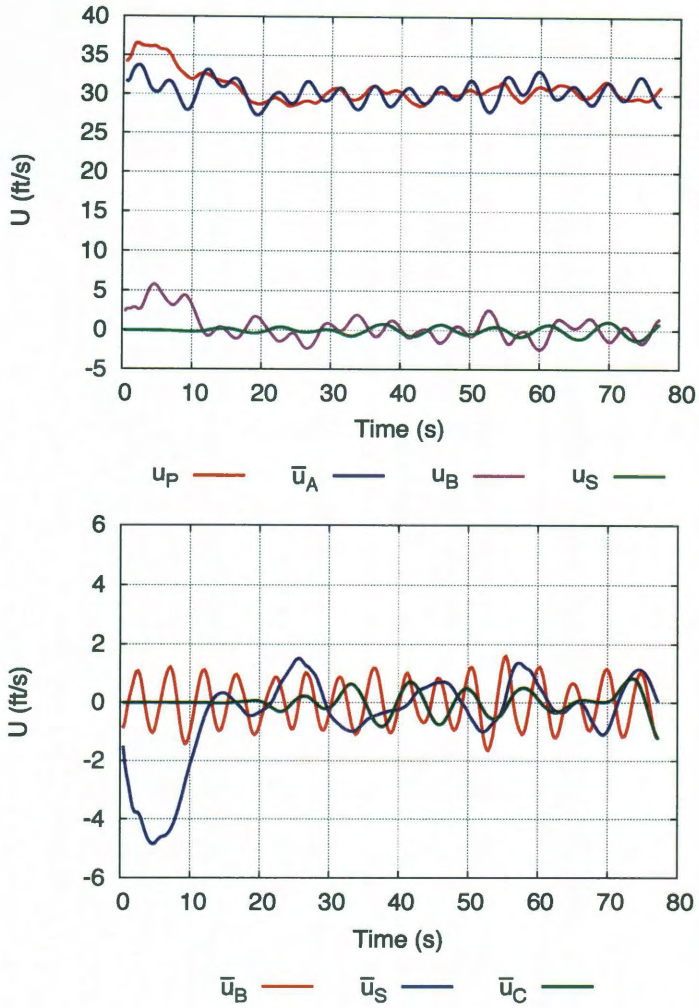


Figure 7.5: Decomposition of the descent speed for the cluster computation with PLC and $\theta_{\text{INIT}} = 35°$.

area to the freestream. Therefore, Parachute 1 produces substantially more drag than Parachute 2 as shown in Figure 7.13. At the time of maximum drag differential, Parachute 1 produces nearly 70% of the total cluster drag as shown in Figure 7.14.

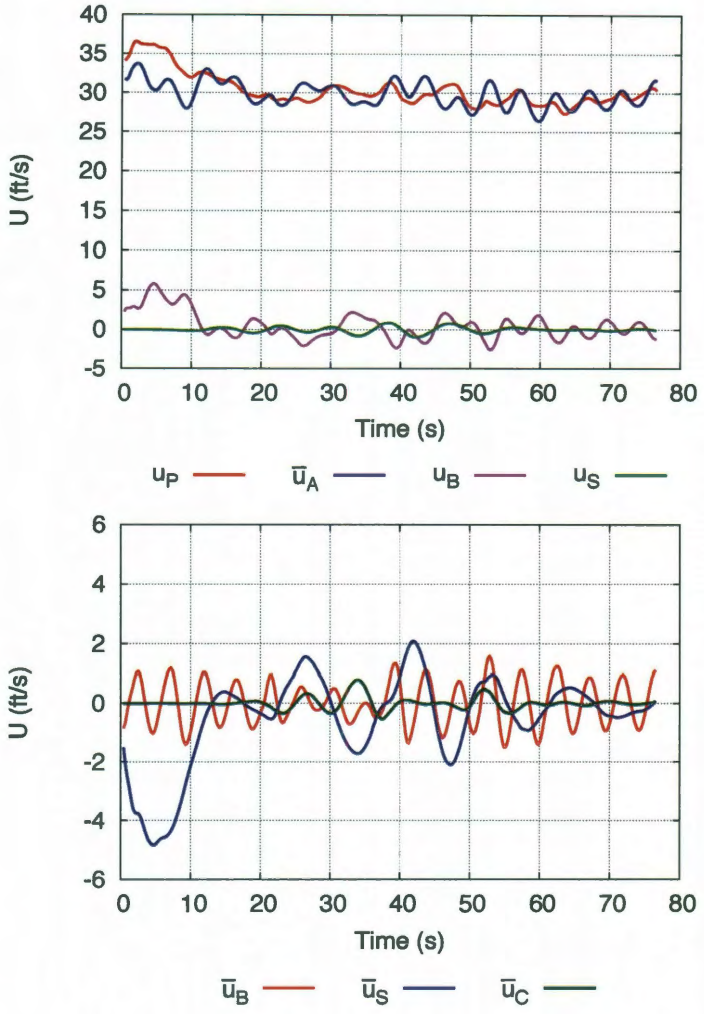


Figure 7.6: Decomposition of the descent speed for the cluster computation with PTE and $\theta_{\text{INIT}} = 35°$.

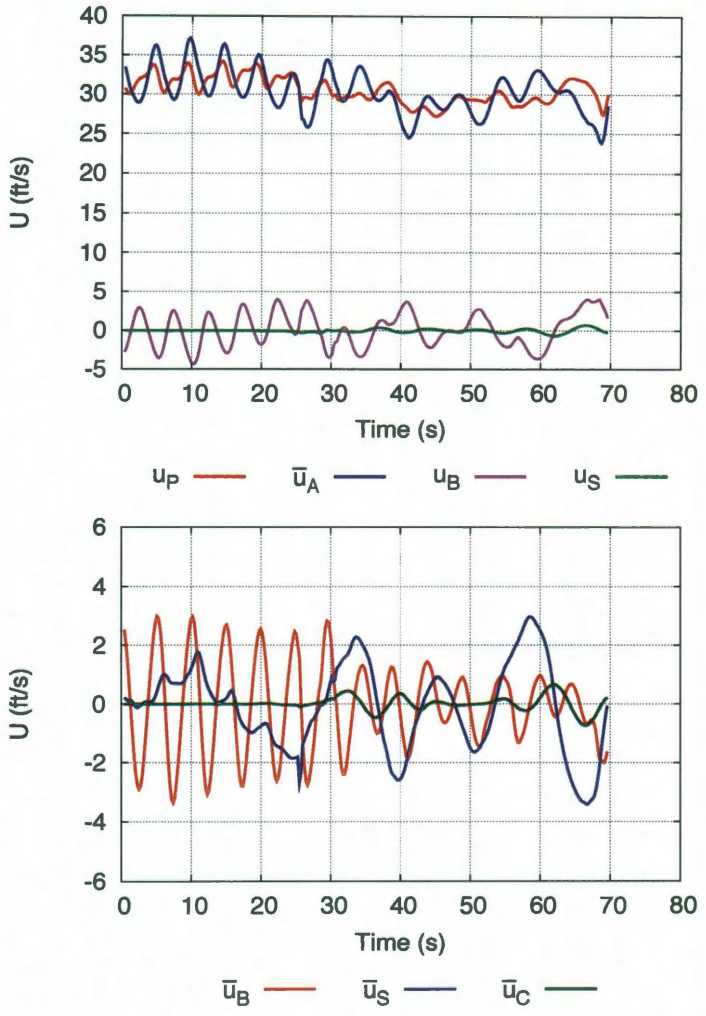


Figure 7.7: Decomposition of the descent speed for the cluster computation with PTE and $\theta_{\text{INIT}} = 15°$.

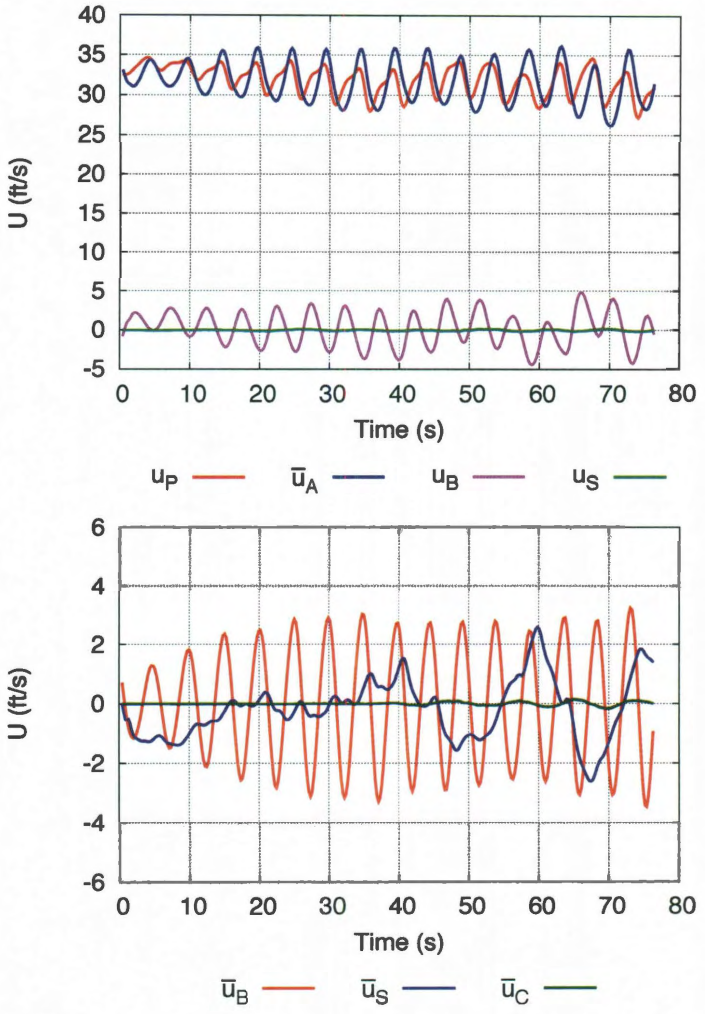


Figure 7.8: Decomposition of the descent speed for the cluster computation with PTE and $\theta_{\text{INIT}} = 25°$.

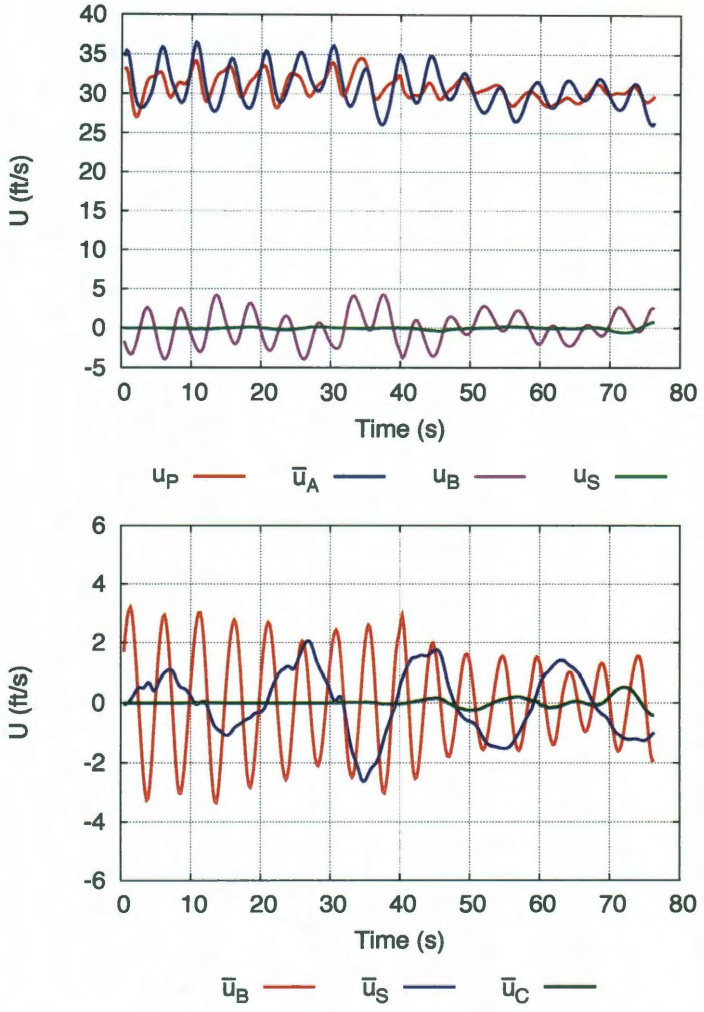


Figure 7.9: Decomposition of the descent speed for the simulated-disreef cluster computation.

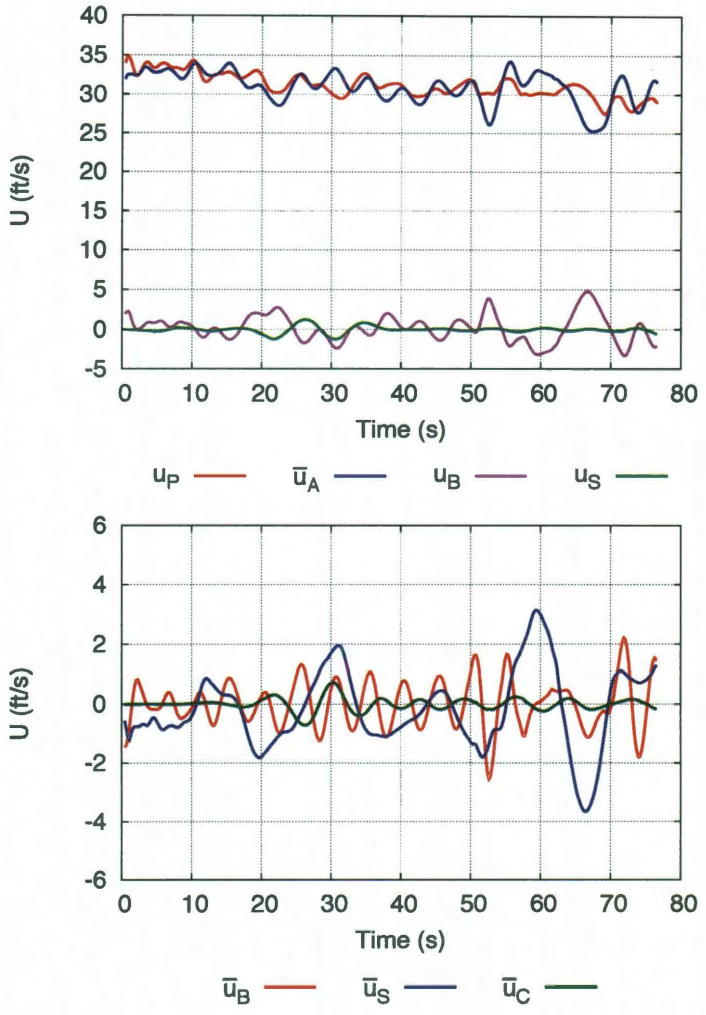


Figure 7.10: Decomposition of the descent speed for the asynchronous-disreef cluster computation.

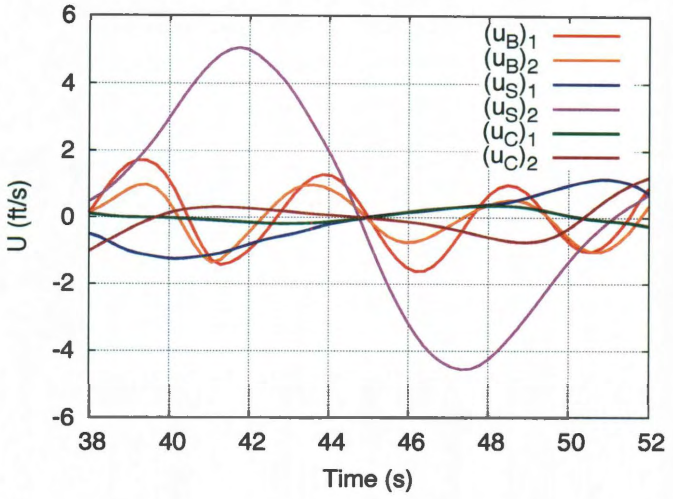


Figure 7.11: Individual-parachute contributions to descent speed for the cluster computation with PTE and $\theta_{\text{INIT}} = 35°$.

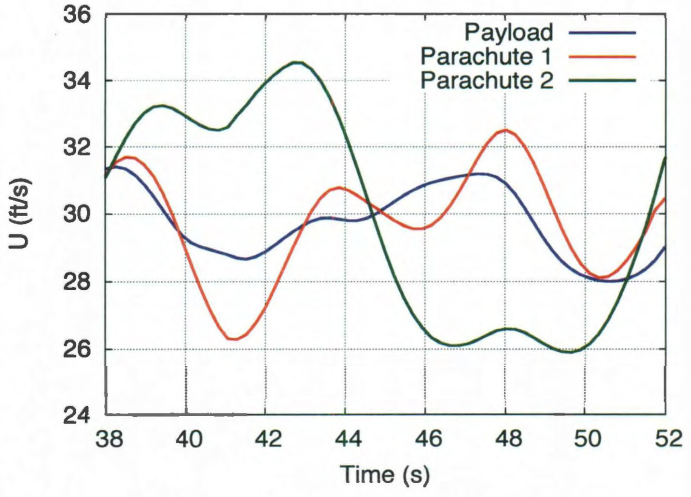


Figure 7.12: Payload and canopy-centroid descent speeds for the cluster computation with PTE and $\theta_{\text{INIT}} = 35°$.

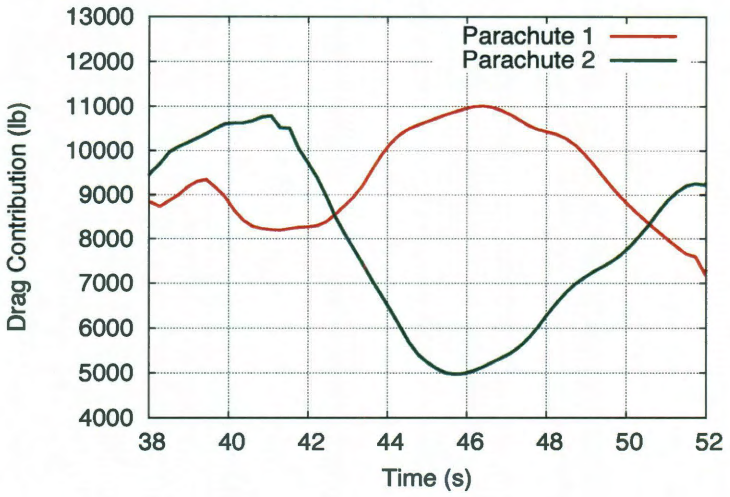


Figure 7.13: Drag contribution of each parachute for the cluster computation with PTE and $\theta_{\text{INIT}} = 35°$.

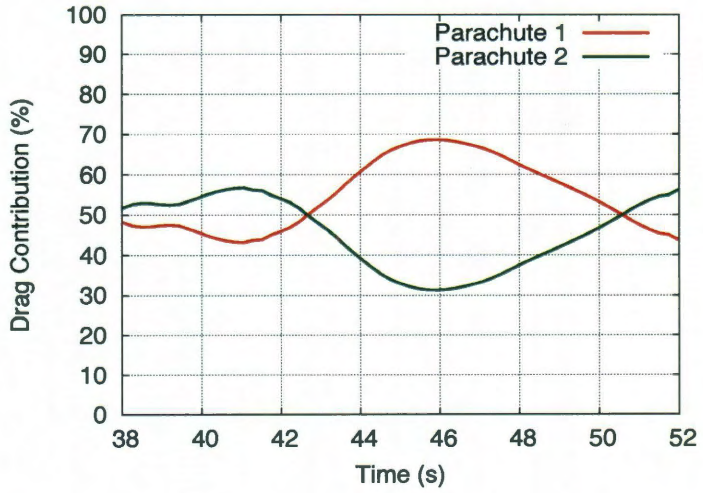


Figure 7.14: Drag contribution percentage of each parachute for the cluster computation with PTE and $\theta_{\text{INIT}} = 35°$.

7.2 Added Mass

7.2.1 Concept

Added mass is the inertia added to a system due to the unsteady motion of a body through a surrounding fluid. The concept of added mass is applicable to parachutes during disreefing and unsteady full-open descent. When the parachute is disreefed, the sudden increase in drag area causes a rapid deceleration and increase in system inertia. During full-open descent, the parachute exhibits a periodic breathing motion. The frequency and damping of this motion is related to the added mass. Here we present a technique for determining the added mass of a parachute from FSI computations, and we approximate the added mass of a single parachute in full-open descent.

7.2.2 Theoretical Background

In a vacuum, the mass of an accelerating body can be known precisely. However, in a fluid medium, the determination of the effective mass of a body requires the consideration of the motion of the fluid as well as that of the body. When a body immersed in a fluid translates at a steady velocity U, the body experiences a drag force D, which is proportional to U^2:

$$D = \frac{1}{2}\rho S C_{\mathrm{D}} U^2, \tag{7.39}$$

where S is a reference area. If the body accelerates, there will be an increase in the drag associated with the instantaneous value of U. The fluid surrounding the body will also acquire momentum and the kinetic energy of the fluid will increase. This energy must be supplied in the form of work done on the fluid by the body. As a result, the body will experience an additional drag force. It is sometimes convenient to think of this additional force in terms of an added mass of fluid that is being

accelerated along with the body. The instantaneous force on the body at time t can then be expressed as

$$F(t) = D + m_{\mathrm{A}}\dot{U}(t), \tag{7.40}$$

where D, as before, is the instantaneous drag as a function of U^2, m_{A} is the added mass, and $\dot{U}(t)$ is the instantaneous acceleration of the body.

7.2.3 Determining Added Mass

Rewriting Eq. (7.40) in a general form, the force experienced by a body moving through a fluid is a function of the instantaneous velocity and instantaneous acceleration of the body. The force at time t can be written in terms of coefficients C and m_{A} as

$$F(t) = CU_{\mathrm{A}}^2(t) + m_{\mathrm{A}}\dot{U}_{\mathrm{A}}(t), \tag{7.41}$$

where m_{A} is the added mass. Dividing both sides of Eq. (7.41) by $U_{\mathrm{A}}^2(t)$, we obtain

$$\frac{F(t)}{U_{\mathrm{A}}^2(t)} = C + m_{\mathrm{A}}\frac{\dot{U}_{\mathrm{A}}(t)}{U_{\mathrm{A}}^2(t)}. \tag{7.42}$$

Plotting data points from parachute FSI computations in this form and fitting a trend line to the data can yield values for C and m_{A}.

Figure 7.15 shows two breathing periods from a symmetric FSI computation for a single parachute in full-open descent. The parachute exhibits a periodic breathing motion resulting in a time-varying added mass. Figure 7.16 shows the projected area of the parachute for the time steps corresponding to the data points and the colors illustrate the chronological sequence of the data points during the breathing period.

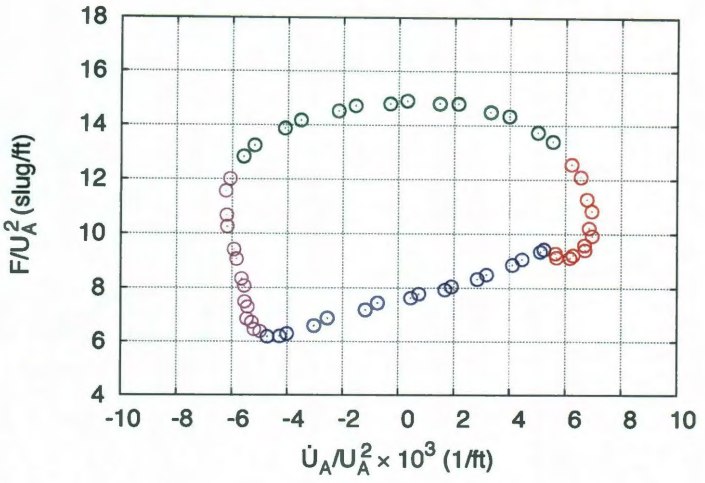


Figure 7.15: Symmetric FSI computation for a single parachute in full-open descent. Results are plotted in the form of Eq. (7.42). Two breathing periods are shown.

Figure 7.17 shows the full results from the same symmetric FSI computation for approximately 150 s of full-open descent. Linear regression is used to fit a trend line to the part of the periodic motion when the parachute diameter is increasing. From the plot we find $m_{\mathrm{A}} = 339$ slugs and $C = 7.7$ slugs/ft. Added mass during the inflation part of the breathing period is nearly constant at approximately 10,900 lbs. This is almost twice the payload weight of 5,570 lbs and approximately 10% larger than the enclosed air mass. Some parts of the breathing period have zero slope indicating that there is no added mass. Other parts of the breathing period have a negative slope. In a physical sense, this means that the fluid is doing work on the parachute.

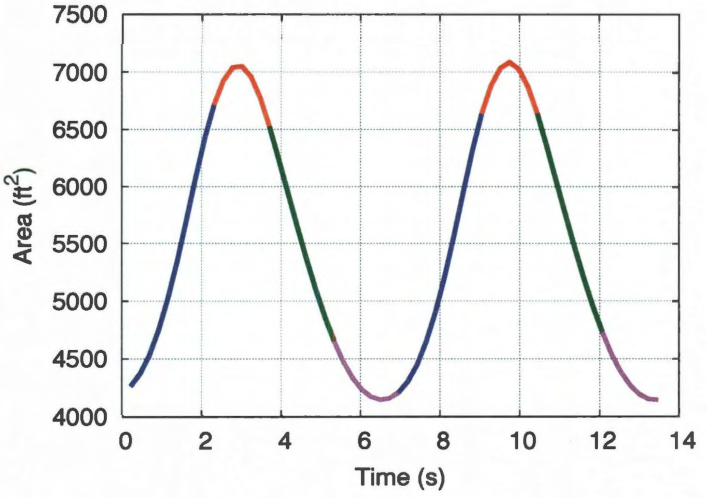


Figure 7.16: Parachute projected area for the time steps corresponding to the data points shown in Figure 7.15.

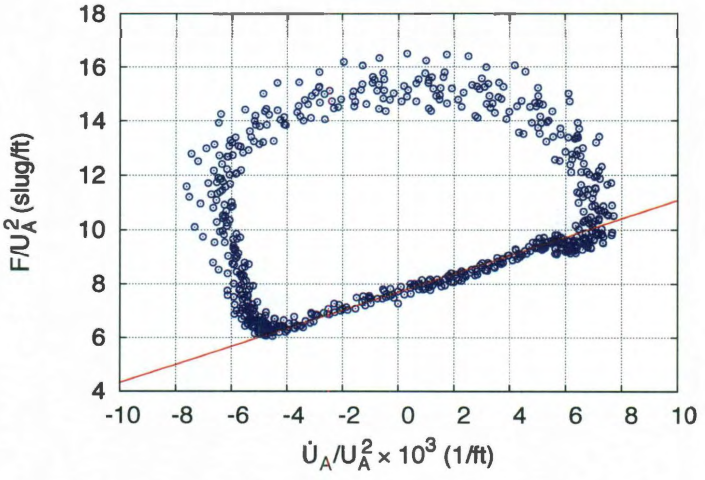


Figure 7.17: Full results for the single parachute symmetric FSI computation. The data points shown here cover approximately 150 s of full-open descent.

Chapter 8

Preliminary Disreef Computations

The Orion main parachutes employ a reefing technique during deployment to permit incremental opening of the parachute and to protect the integrity of the canopy fabric. When the parachute initially exits the deployment bag, the skirt is constricted by reefing lines to prevent the canopy from fully opening. At preselected time intervals, the reefing lines are automatically cut and the parachute is "disreefed" to a larger diameter. Parachute reefing serves at least two main purposes [16]. First, it limits the peak loads on the fabric, radials, and suspension lines by opening the parachute in stages. Consequently, this also limits the acceleration forces felt by the occupants of the capsule. Secondly, reefing increases the accuracy of drops from high altitude by decreasing drag area and allowing a high rate of descent. Higher descent speed translates to decreased horizontal gliding tendencies. Shortly before reaching the water, the parachute is disreefed to achieve a safe impact velocity.

The amount by which the parachute skirt is constrained is described by a parameter known as the reefing-line ratio which is defined as $\tau_{\text{REEF}} = D_{\text{REEF}}/D_0$ where D_{REEF} is the reefed diameter and D_0 is the nominal diameter [16]. The Orion parachutes have two reefing stages. Immediately after deployment, the parachute is in the Stage 1 configuration with $\tau_{\text{REEF}} = 7\%$. After a preselected time interval, the parachute is

disreefed to Stage 2 with $\tau_{\text{REEF}} = 14\%$. After another preselected time interval, the parachute is disreefed from Stage 2 to the full-open configuration.

8.1 Starting Conditions

Parachute reefing is accomplished incrementally with a series of stand-alone structural mechanics computations. We start with a full-open canopy shape from a single parachute FSI computation and begin reefing at a point in the periodic breathing cycle when the diameter is at a minimum. During the first computation, the reefing-line element lengths are decreased in a linear fashion until $\tau_{\text{REEF}} = 50\%$. Simultaneously, a circumferentially-uniform and time-varying prescribed pressure profile is applied to the structural mechanics mesh. The initial pressure profile comes from a full-open symmetric FSI computation. The final pressure profile comes from earlier T★AFSM computations at corresponding τ_{REEF} values [55]. Those earlier computations used a slightly different version of the parachute structural mechanics mesh. The forces applied to the structure are ramped from the initial to the final pressure profile over a period of 45 s using a Cosine form. The same procedure is used to reef the parachute from 50% to 25%, and from 25% to 14%. The pressure profile corresponding to the Stage 2 shape is held steady until the structure reaches a settled shape. A fluid mechanics mesh is generated around the settled starting shape. The developed flow field is obtained with a two-part stand-alone fluid mechanics computation, similar to the way it was described in Section 5.2, based on the starting descent speed. After obtaining a starting shape and developed flow field, we begin a symmetric FSI computation.

At the beginning of the symmetric FSI computation, the parachute adjusts to the flow conditions and reaches a new settled shape and descent speed. In order to match the conditions observed during Stage 2 descent, we go through an iterative process

during symmetric FSI and obtain a desirable shape and descent speed. We observe in drop test videos that the lower sails of the parachute have significant deformation that allow increased air flow through the lower portion of the canopy. The structural mechanics mesh in the computation does not have enough resolution to model this deformation, and therefore the A_G value from Eq. (4.14) is too small. We cannot change A_G, but we can change the geometric porosity coefficient k_G and achieve the same effect by increasing the cross-canopy velocity due to geometric porosity. In our preliminary testing we have found that increasing k_G to three times the full-open value for the lower five sails results in a parachute shape and descent speed that closely matches drop tests. The final settled shape is shown in Figure 8.1.

Figure 8.1: Parachute structural mesh in the Stage 2 configuration.

8.2 Computational Conditions

The computational conditions for the disreefing case are the same as those for the single parachute parametric study described in Section 5.2 with a few exceptions, which we describe here. The starting descent speed is 62.5 ft/s, which is the estimated descent speed for the Stage 2 configuration. The initial fluid mechanics mesh (before any remeshing) has 210,807 nodes and 1,304,389 four-node tetrahedral elements. We do not use SSP for reefed parachute computations because we have found that projecting the fluid tractions as vectors, which are normal to the fluid interface, helps to maintain a good reefed shape and preserve the quality of the fluid mechanics mesh. The structure part of the selective scaling is set to 1000. The time-step size is 0.01 s, and the number of GMRES iterations per nonlinear iteration is 180 for the fluid+structure block. The porosity is modeled using a modified version of HMGP-FG that was developed to address the additional geometric complexity caused by the reefed parachute shape. The modified version of HMGP-FG is described in Section 8.3.

8.3 HMGP-FG Regularization

From Eq. (4.14), the velocity of cross-canopy flow due to geometric porosity can be expressed as

$$(u_n)_G = -(k_G)_J \frac{A_G}{A_1} sgn(\Delta p) \sqrt{\frac{|\Delta p|}{\rho}}. \tag{8.1}$$

In the method we use to solve the nonlinear equation system in the fluid mechanics problem, we must calculate the partial derivative of $(u_n)_G$ with respect to Δp. That

partial derivative can be expressed as

$$\frac{\partial (u_n)_G}{\partial \Delta p} = -(k_G)_J \frac{A_G}{A_1} \frac{1}{2} \sqrt{\frac{1}{\rho |\Delta p|}}. \tag{8.2}$$

Note that a discontinuity exists at $\Delta p = 0$. We have also found that when Δp is very small, Eq. (8.1) does not accurately model the relationship between Δp and $(u_n)_G$ that we observe in higher resolution fluid mechanics computations like the ones mentioned in Section 4.3.1.

In the case of full-open parachute computations, the flow separates at the parachute skirt and Δp is sufficiently high for all regions of the canopy. However, in the case of a reefed parachute, the flow separates approximately five sails above the skirt. In that separation region, Δp varies between positive and negative values. This causes numerical instability in the fluid part of the solution. To address this challenge, we modify Eq. (8.1) as follows:

$$(u_n)_G = -(k_G)_J \frac{A_G}{A_1} sgn(\Delta p) \left(\sqrt{\frac{|\Delta p| + \epsilon_p}{\rho}} - \sqrt{\frac{\epsilon_p}{\rho}} \right). \tag{8.3}$$

where ϵ_p is a parameter in units of pressure. We call this new version HMGP-FGR, where "R" stands for regularized. Note that when $\Delta p = 0$, there is no cross-canopy velocity. We calibrate the value of ϵ_p using data from a 4-gore fluid mechanics computation. If ϵ_p is sufficiently small, we can avoid numerical instability and still maintain solution accuracy. For preliminary reefing and disreef computations, we have used $\epsilon_p / (\rho \nu^2 / A_G) = 812.9$. Figure 8.2 shows the HMGP-FGR model compared with data points from a 4-gore fluid mechanics computation.

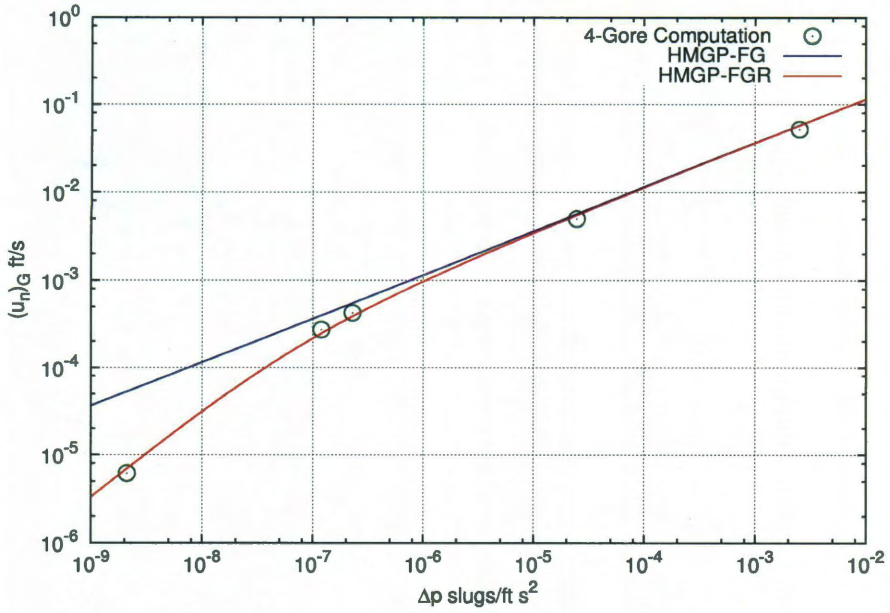


Figure 8.2: Comparison of HMGP-FG and HMGP-FGR with data from a 4-gore fluid mechanics computation. Data shown is for Patch 5. This comparison uses $\epsilon_p/(\rho\nu^2/A_G) = 812.9$.

8.4 Preliminary Results

After obtaining the desired parachute shape and descent speed using the techniques described, we begin the de-symmetrization using a Cosine form which lasts for about 7 s. The parachute is then instantaneously disreefed by giving the reefing line elements zero-stiffness. As the parachute opens, we observe deformation on top of the canopy near the vent due to wake recontact (see [23] for this parachute behavior). Figure 8.3 shows the disreef sequence from the FSI computation. We consider these results to be preliminary.

Figure 8.3: Results from FSI computation of parachute disreefing from Stage 2 to the full-open configuration. The top left picture shows the instant that disreefing starts, and the remaining pictures proceed in 0.22 s intervals from left to right and top to bottom.

Chapter 9

Conclusions

Space–time FSI techniques and other special techniques developed by the T★AFSM have been used to address some of the numerical challenges associated with computer modeling of spacecraft parachutes and parachute clusters. We use these techniques for computer modeling of the parachute system for NASA's Orion CEV Crew Module. This type of computational approach has the potential to reduce development costs, model the parachute in conditions that are not possible to achieve in a laboratory, and allow in-depth analysis to complement field testing.

Several special techniques that have been developed for single parachute and parachute cluster computations have been described in Chapter 4. To shelter the fluid mechanics mesh from the geometric complexities of the ringsail parachute geometry, we use the FSI-DGST and carry out geometric smoothing in the circumferential direction of the parachute canopy. In the SSP technique, the pressure and viscous parts of the stress at the fluid interface are projected to the structure interface separately. Pressure is projected as a scalar and viscous stress as a vector. With HMGP, we model the combination of geometric and fabric porosity as an equivalent, locally-varying fabric porosity. In an improvement to HMGP called HMGP-FG, we separate the fabric and geometric contributions. The velocity coming from the

fabric contribution varies linearly with pressure differential, and the velocity coming from the geometric contribution varies nonlinearly. We use symmetric FSI to build a good starting condition for the full FSI computation without generating asymmetric parachute deformation or gliding in the process.

A single-parachute FSI computation has been described in Chapter 5. The computation analyzes the effect of design variations on parachute performance. The design variations include increased canopy loading and the addition of an OICL to decrease descent speed oscillations. A method for removing the swinging component of the payload descent speed has been described. The method is intended to provide a better comparison basis in parametric studies. Results for the single-parachute computation have been reported including descent and total speeds, parachute diameter, and L/D ratio.

Parachute cluster FSI computations have been presented in Chapter 6. The computations are part of an analysis to determine how the parameters representing payload models and starting conditions affect long-term cluster dynamics. The parameters selected for testing are the payload-model configurations and initial coning angles (θ_{INIT}) and parachute diameters (D_{INIT}). Two scenarios to approximate the conditions immediately after disreefing have also been investigated. Payload descent speed, drag coefficient, and vent-separation distance have been reported.

A technique for modeling the contact between parachutes in a cluster has also been described in Chapter 6. The contact algorithm is based on preventing the structural surfaces from coming closer than a predetermined minimum distance to protect the quality of the fluid mechanics mesh between the structural surfaces. In the latest version, called SENCT-FC, the forces are applied in a conservative fashion. Using the "-M1" option, the contact model is formulated based on the fluid mechanics mesh at the interface. This option is particularly helpful when smoothing and homogenization techniques are employed.

Dynamical analysis techniques for parachute FSI computations have been presented in Chapter 7. The critical measure of performance for the Orion parachute system is the descent speed of the payload, and one of the goals of the parachute cluster computational analysis is to assist design engineers in determining which factors contribute to the payload descent speed oscillations. For that, a special decomposition technique developed by the T★AFSM was used for parachute descent speed. In addition to the time histories of the aerodynamically significant quantities, such as the descent velocity and forces, the various contributors to these quantities have been shown separately. A special technique has also been presented for extracting model parameters from a parachute FSI computation. These parameters, including the added mass, can be used in fast, approximate engineering analysis models for parachute dynamics.

Chapter 8 describes a single-parachute disreef computation and the techniques used to address specific challenges associated with reefed parachute computations. The methods and computations used for developing a starting parachute shape and flowfield have been described. A method to address numerical challenges related to the reefed parachute problem by regularizing HMGP-FG has also been introduced. Preliminary disreef computational results have been presented.

Bibliography

[1] J. E. Akin, T. Tezduyar, M. Ungor, and S. Mittal. Stabilization parameters and Smagorinsky turbulence model. *Journal of Applied Mechanics*, 70:2–9, 2003.

[2] J. E. Akin and T. E. Tezduyar. Calculation of the advective limit of the SUPG stabilization parameter for linear and higher-order elements. *Computer Methods in Applied Mechanics and Engineering*, 193:1909–1922, 2004.

[3] R. J. Benney, K. R. Stein, J. W. Leonard, and M. L. Accorsi. Current 3-D structural dynamic finite element modeling capabilities. In *Proceedings of AIAA 14th Aerodynamic Decelerator Systems Technology Conference*, AIAA Paper 97-1506, San Francisco, California, 1997.

[4] L. Catabriga, A. L. G. A. Coutinho, and T. E. Tezduyar. Compressible flow SUPG parameters computed from element matrices. *Communications in Numerical Methods in Engineering*, 21:465–476, 2005.

[5] L. Catabriga, A. L. G. A. Coutinho, and T. E. Tezduyar. Compressible flow SUPG parameters computed from degree-of-freedom submatrices. *Computational Mechanics*, 38:334–343, 2006.

[6] U. S. Congress. NASA Authorization Act of 2010. `http://www.gpo.gov/fdsys/pkg/BILLS-111s3729enr/pdf/BILLS-111s3729enr.pdf`.

[7] A. Corsini, F. Rispoli, and T. E. Tezduyar. Stabilized finite element computation of NOx emission in aero-engine combustors. *International Journal for Numerical Methods in Fluids*, 65:254–270, 2011.

[8] T. Fujisawa, M. Inaba, and G. Yagawa. Parallel computing of high-speed compressible flows using a node-based finite element method. *International Journal for Numerical Methods in Fluids*, 58:481–511, 2003.

[9] H. M. Hilber, T. J. R. Hughes, and R. L. Taylor. Improved numerical dissipation for time integration algorithms in structural dynamics. *Earthquake Engineering and Structural Dynamics*, 5:283–292, 1977.

[10] M.-C. Hsu, Y. Bazilevs, V. M. Calo, T. E. Tezduyar, and T. J. R. Hughes. Improving stability of stabilized and multiscale formulations in flow simulations at small time steps. *Computer Methods in Applied Mechanics and Engineering*, 199:828–840, 2010.

[11] Z. Johan, K. K. Mathur, S. L. Johnsson, and T. J. R. Hughes. A case study in parallel computation: Viscous flow around an Onera M6 wing. *International Journal for Numerical Methods in Fluids*, 21:877–884, 1995.

[12] A. A. Johnson and T. E. Tezduyar. Mesh update strategies in parallel finite element computations of flow problems with moving boundaries and interfaces. *Computer Methods in Applied Mechanics and Engineering*, 119:73–94, 1994.

[13] A. A. Johnson and T. E. Tezduyar. Simulation of multiple spheres falling in a liquid-filled tube. *Computer Methods in Applied Mechanics and Engineering*, 134:351–373, 1996.

[14] V. Kalro and T. E. Tezduyar. A parallel 3D computational method for fluid–structure interactions in parachute systems. *Computer Methods in Applied Mechanics and Engineering*, 190:321–332, 2000.

[15] G. Karypis and V. Kumar. A fast and high quality multilevel scheme for partitioning irregular graphs. *SIAM Journal of Scientific Computing*, 20:359–392, 1998.

[16] T. W. Knacke. *Parachute Recovery Systems: Design Manual.* Para Publishing, Santa Barbara, CA, 1992.

[17] A. Lo. *Nonlinear Dynamic Analysis of Cable and Membrane Structure.* PhD thesis, Department of Civil Engineering, Oregon State University, 1982.

[18] C. J. Moorman. *Fluid–Structure Interaction Modeling of the Orion Spacecraft Parachutes.* PhD thesis, Rice University, 2010.

[19] NASA. The Vision for Space Exploration, 2004. `http://www.nasa.gov/pdf/55583main_vision_space_exploration2.pdf`.

[20] NASA. Crew Module/Orion Project Office, 2009. `http://spaceflightsystems.grc.nasa.gov/Orion/CrewModule/`.

[21] NASA. Orion: America's Next Generation Spacecraft, 2009. `http://www.nasa.gov/pdf/491544main_orion_book_web.pdf`.

[22] NASA. Orion Crew Exploration Vehicle, 2009. `http://www.nasa.gov/pdf/306407main_orion_crew_expl_vehicle.pdf`.

[23] C. W. Peterson, J. H. Strickland, and H. Higuchi. The Fluid Dynamics of Parachute Inflation. *Annual Review of Fluid Mechanics*, 28:361–387, 1996.

[24] F. Rispoli, A. Corsini, and T. E. Tezduyar. Finite element computation of turbulent flows with the discontinuity-capturing directional dissipation (DCDD). *Computers & Fluids*, 36:121–126, 2007.

[25] Y. Saad and M. Schultz. GMRES: A generalized minimal residual algorithm for solving nonsymmetric linear systems. *SIAM Journal of Scientific and Statistical Computing*, 7:856–869, 1986.

[26] S. Sathe and T. E. Tezduyar. Modeling of fluid–structure interactions with the space–time finite elements: Contact problems. *Computational Mechanics*, 43:51–60, 2008.

[27] L. Singleton. Orion Spacecraft on the Path to Future Flight, 2010. `http://www.lockheedmartin.com/news/press_releases/2010/0921_ss_orion.html`.

[28] K. Stein, R. Benney, V. Kalro, T. E. Tezduyar, J. Leonard, and M. Accorsi. Parachute fluid–structure interactions: 3-D Computation. *Computer Methods in Applied Mechanics and Engineering*, 190:373–386, 2000.

[29] K. Stein, T. Tezduyar, and R. Benney. Mesh moving techniques for fluid–structure interactions with large displacements. *Journal of Applied Mechanics*, 70:58–63, 2003.

[30] K. Stein, T. E. Tezduyar, and R. Benney. Automatic mesh update with the solid-extension mesh moving technique. *Computer Methods in Applied Mechanics and Engineering*, 193:2019–2032, 2004.

[31] K. Takizawa, J. Christopher, T. E. Tezduyar, and S. Sathe. Space–time finite element computation of arterial fluid–structure interactions with patient-specific data. *International Journal for Numerical Methods in Biomedical Engineering*, 26:101–116, 2010.

[32] K. Takizawa, C. Moorman, S. Wright, T. Spielman, and T. E. Tezduyar. Fluid–structure interaction modeling and performance analysis of the Orion spacecraft parachutes. *International Journal for Numerical Methods in Fluids*, 65:271–285, 2011.

[33] K. Takizawa, T. Spielman, and T. E. Tezduyar. Space–time FSI modeling and dynamical analysis of spacecraft parachutes and parachute clusters. *Computational Mechanics*, published online, DOI: 10.1007/s00466-011-0590-9, April 2011.

[34] K. Takizawa, S. Wright, C. Moorman, and T. E. Tezduyar. Fluid–structure interaction modeling of parachute clusters. *International Journal for Numerical Methods in Fluids*, 65:286–307, 2011.

[35] T. Tezduyar, S. Aliabadi, M. Behr, A. Johnson, and S. Mittal. Parallel finite-element computation of 3D flows. *Computer*, 26(10):27–36, 1993.

[36] T. E. Tezduyar. Stabilized finite element formulations for incompressible flow computations. *Advances in Applied Mechanics*, 28:1–44, 1992.

[37] T. E. Tezduyar. Finite element methods for flow problems with moving boundaries and interfaces. *Archives of Computational Methods in Engineering*, 8:83–130, 2001.

[38] T. E. Tezduyar. Computation of moving boundaries and interfaces and stabilization parameters. *International Journal for Numerical Methods in Fluids*, 43:555–575, 2003.

[39] T. E. Tezduyar. Finite element methods for fluid dynamics with moving boundaries and interfaces. In E. Stein, R. D. Borst, and T. J. R. Hughes, editors, *Encyclopedia of Computational Mechanics*, Volume 3: Fluids, chapter 17. John Wiley & Sons, 2004.

[40] T. E. Tezduyar. Finite elements in fluids: Special methods and enhanced solution techniques. *Computers & Fluids*, 36:207–223, 2007.

[41] T. E. Tezduyar. Finite elements in fluids: Stabilized formulations and moving boundaries and interfaces. *Computers & Fluids*, 36:191–206, 2007.

[42] T. E. Tezduyar, M. Behr, and J. Liou. A new strategy for finite element computations involving moving boundaries and interfaces – the deforming-spatial-domain/space–time procedure: I. The concept and the preliminary numerical tests. *Computer Methods in Applied Mechanics and Engineering*, 94(3):339–351, 1992.

[43] T. E. Tezduyar, M. Behr, S. Mittal, and A. A. Johnson. Computation of unsteady incompressible flows with the finite element methods – space–time formulations, iterative strategies and massively parallel implementations. In *New Methods in Transient Analysis*, PVP-Vol.246/AMD-Vol.143, pages 7–24, New York, 1992. ASME.

[44] T. E. Tezduyar, M. Behr, S. Mittal, and J. Liou. A new strategy for finite element computations involving moving boundaries and interfaces – the deforming-spatial-domain/space–time procedure: II. Computation of free-surface flows, two-liquid flows, and flows with drifting cylinders. *Computer Methods in Applied Mechanics and Engineering*, 94(3):353–371, 1992.

[45] T. E. Tezduyar and Y. Osawa. Finite element stabilization parameters computed from element matrices and vectors. *Computer Methods in Applied Mechanics and Engineering*, 190:411–430, 2000.

[46] T. E. Tezduyar and S. Sathe. Modeling of fluid–structure interactions with the space–time finite elements: Solution techniques. *International Journal for Numerical Methods in Fluids*, 54:855–900, 2007.

[47] T. E. Tezduyar, S. Sathe, R. Keedy, and K. Stein. Space–time techniques for finite element computation of flows with moving boundaries and interfaces. In S. Gallegos, I. Herrera, S. Botello, F. Zarate, and G. Ayala, editors, *Proceedings of*

the III International Congress on Numerical Methods in Engineering and Applied Science. CD-ROM, Monterrey, Mexico, 2004.

[48] T. E. Tezduyar, S. Sathe, R. Keedy, and K. Stein. Space–time finite element techniques for computation of fluid–structure interactions. *Computer Methods in Applied Mechanics and Engineering*, 195:2002–2027, 2006.

[49] T. E. Tezduyar, S. Sathe, J. Pausewang, M. Schwaab, J. Christopher, and J. Crabtree. Fluid–structure interaction modeling of ringsail parachutes. *Computational Mechanics*, 43:133–142, 2008.

[50] T. E. Tezduyar, S. Sathe, J. Pausewang, M. Schwaab, J. Christopher, and J. Crabtree. Interface projection techniques for fluid–structure interaction modeling with moving-mesh methods. *Computational Mechanics*, 43:39–49, 2008.

[51] T. E. Tezduyar, S. Sathe, J. Pausewang, M. Schwaab, J. Crabtree, and J. Christopher. Air–fabric interaction modeling with the Stabilized Space–Time FSI technique. In *Proceedings of the Third Asian-Pacific Congress on Computational Mechanics (CD-ROM)*, Kyoto, Japan, 2007.

[52] T. E. Tezduyar, S. Sathe, J. Pausewang, M. Schwaab, J. Crabtree, and J. Christopher. Fluid-structure interaction modeling with moving-mesh techniques. In *Proceedings of the Symposium on Recent Progress in Computational Fluid Dynamics, Japan Society of Automotive Engineers*, Tokyo, Japan, 2007.

[53] T. E. Tezduyar, S. Sathe, and K. Stein. Solution techniques for the fully-discretized equations in computation of fluid–structure interactions with the space–time formulations. *Computer Methods in Applied Mechanics and Engineering*, 195:5743–5753, 2006.

[54] T. E. Tezduyar, K. Takizawa, T. Brummer, and P. R. Chen. Space–time fluid–structure interaction modeling of patient-specific cerebral aneurysms. *Interna-*

tional Journal for Numerical Methods in Biomedical Engineering, published online, DOI: 10.1002/cnm.1433, February 2011.

[55] T. E. Tezduyar, K. Takizawa, C. Moorman, S. Wright, and J. Christopher. Space–time finite element computation of complex fluid–structure interactions. *International Journal for Numerical Methods in Fluids*, 64:1201–1218, 2010.